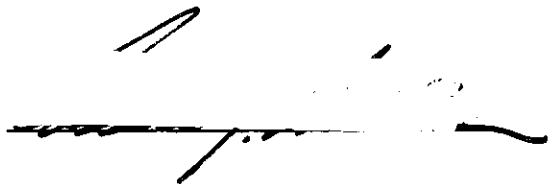


In presenting the dissertation as a partial fulfillment of the requirements for an advanced degree from the Georgia Institute of Technology, I agree that the Library of the Institute shall make it available for inspection and circulation in accordance with its regulations governing materials of this type. I agree that permission to copy from, or to publish from, this dissertation may be granted by the professor under whose direction it was written, or, in his absence, by the Dean of the Graduate Division when such copying or publication is solely for scholarly purposes and does not involve potential financial gain. It is understood that any copying from, or publication of, this dissertation which involves potential financial gain will not be allowed without written permission.

A handwritten signature in dark ink, consisting of a series of fluid, connected strokes. The signature is positioned to the right of the main text block.

7/25/68

SIMPLEX EVOLUTIONARY OPERATION FOR IMPROVEMENT
OF TRAFFIC SIGNAL SETTINGS

A THESIS

Presented to

The Faculty of the Graduate Division

by

Cassius J. Mullen

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in the School of Industrial and Systems

Engineering

Georgia Institute of Technology

June, 1970

SIMPLEX EVOLUTIONARY OPERATION FOR IMPROVEMENT
OF TRAFFIC SIGNAL SETTINGS

Approved:

Chairman

Date approved by Chairman: May 19, 1970

ACKNOWLEDGMENTS

I would like to express my gratitude to many individuals for help in preparation of this thesis. I thank the members of my reading committee, Dr. Douglas C. Montgomery, Dr. Joseph Talavage, and Dr. John J. Jarvis, for reading the original draft of this thesis for content and form. Dr. Montgomery, my thesis advisor, gave invaluable assistance and recommendations which have certainly resulted in a more meaningful thesis. Dr. Talavage was most helpful in establishing communications with a source of field data for this study.

My thanks to Mr. George S. Scruggs and the Traffic Engineering Department of the City of Wichita Falls, Texas for generously providing realistic field data so necessary for this thesis.

Appreciation is extended to Mr. Michael P. Deisenroth for his aiding in the development of the traffic simulator for this study.

Credit is also due to Mrs. Jean Williams for her skillful assistance in typing and editing the contents of this thesis.

Finally, to my wife, Betty, I extend my most loving appreciation for her patience, understanding, and encouragement which contributed greatly to the completion of this thesis.

TABLE OF CONTENTS

	Page
ACKNOWLEDGMENTS	ii
LIST OF TABLES	v
LIST OF ILLUSTRATIONS	v
Chapter	
I. INTRODUCTION	1
Description of the Problem	
Objective	
Assumptions	
Definition of Terms	
System Evaluation	
II. LITERATURE SURVEY	9
III. A TECHNIQUE FOR IMPROVING SIGNAL SETTINGS	16
The Simplex Design	
The Simplex EVOP Procedure	
Employment of Simplex EVOP	
The Simulation Model	
IV. THE CONDUCT OF THE EXPERIMENT	26
Discussion	
The Experiment	
V. DISCUSSION OF RESULTS	40
The Results	
Vehicle Operating Cost Reduction	
Comparison with Alternative Schemes	
VI. CONCLUSIONS AND RECOMMENDATIONS	55
Summary of Results	
Conclusions	
Recommendations for Further Study	

TABLE OF CONTENTS (Continued)

	Page
Appendices	
A. COMPUTER LISTING FOR THE SIMPLEX EVOP PROCEDURE.	60
B. COMPUTER LISTING FOR THE TRAFFIC SIMULATOR AND SAMPLE DATA	67
C. THE RESULTS OF PROBLEMS STUDIED	97
BIBLIOGRAPHY	113

LIST OF TABLES

Table		Page
1.	Maximum Age Attainable by a Point	20
2.	Vehicle Arrival Intervals for Periods Tested	28
3.	Starting Signal Settings for Points Tested	32
4.	The Sets of Controllable Variables to be Tested For Each Problem	39
5.	Results of the Pooled t Tests Conducted for the A.M. Period	43

LIST OF ILLUSTRATIONS

Figure		Page
1.	Example of a Two Variable Simplex EVOP	18
2.	Sequential Simplex Procedure Flow Chart	22
3.	Traffic Simulation Network Flow Chart	24
4.	Network Layout for A.M. Peak (7:30 - 8:30).	29
5.	Network Layout for Off-peak (3:30 - 4:30)	30
6.	Network Layout for P.M. Peak (5:00 - 6:00).	31

SUMMARY

A complex problem of today's society is the ever growing congestion in the streets of the nation's urban areas. Traffic congestion can be reduced by vast capital expenditures for new roadway construction and improved public transportation facilities. However, another less costly course of action would be to increase the effectiveness of the current road system. This thesis is directed toward this latter possibility. The approach used is to improve traffic control by means of adjusting signal settings to correspond to the needs of the traffic flow for particular periods of the day.

In order to properly evaluate a street control system, a simulation model of traffic movement was constructed. Further, the optimization of settings for a complex network of traffic signals is in itself a formidable and complex mathematical process. The objective of this thesis is to employ simplex Evolutionary Operation as an automatic search technique for improving settings for a network of traffic signals in order to enhance the performance of the overall traffic flow.

A six intersection network was analyzed. The signal settings in effect at the time traffic counts were taken for three periods of the day were used as starting signal data to be improved. For each period a variety of signal setting combinations were tested.

In all cases studied, simplex Evolutionary Operation improved traffic signal settings for the network. This was verified statistically. An appreciable reduction in costs to motorists resulted. A

comparison was made with the method of steepest ascent. This method also produced a significant improvement in traffic signal settings, but is a more complex technique.

The study demonstrated that the simplex Evolutionary Operation technique warrants being added to the inventory of traffic signal setting improvement schemes currently available.

CHAPTER I

INTRODUCTION

The growing traffic congestion problem in our metropolitan areas may be remedied through the improvement of traffic control. The finding of a near optimal or improved set of signal settings for a signalized street network would constitute a significant approach to the solution of urban traffic congestion. This thesis is directed to that problem.

Description of the Problem

Traffic congestion is encountered in almost all urban areas. This problem shows little sign of impending solution or abatement. Unfortunately this situation will grow even worse with the ever-increasing volume of automobiles in urban areas (1). At present the approaches toward the reduction of traffic congestion are the building of improved roadways, the construction and improvement of mass transit facilities, and the improvement of the utilization of the current road system. The latter approach appears to be the most feasible. First, it may postpone the other two more drastic measures. Secondly, the cost of even the most sophisticated means of traffic control for improvement of roadway utilization is far less than new road or mass transit system construction (2) and (3).

The primary traffic control devices of an urban area are traffic signals which allocate the right of way by alternately directing traffic to stop and proceed. Therefore the signalized traffic intersection may

be considered as the most critical component of the urban traffic network. It is the focal point of congestion and the source of the longest delays. By coordinating the timing or synchronization of a system of signal lights, the flow of traffic can be considerably altered. For purposes of this research, a traffic signal system consists of two or more individual signal installations which operate with a fixed time relationship to each other (1).

Perhaps the ideal way to set traffic signals would be to instrument a street network so as to measure stops and delays continuously, and then by extensive experimentation find the control procedure that maximizes the effectiveness of the system as a whole. However, even when effective experimentation is utilized, it is doubtful that on-street experimentation could search through the tremendous number of control possibilities without an off-street theory to identify relevant choices.

From the standpoint of realism the next best thing to instrumentation would be an accurate digital simulation of traffic movement through a street network. With such a simulation the stops and delays of vehicles can be easily recorded and an evaluation of a traffic control system made.

Of the many possible procedures available, the fixed time system appears to be a realistic control mode to utilize for research purposes (2). First, many cities by virtue of size, traffic conditions, or financial conditions cannot justify an elaborate traffic control system in the immediate future. Secondly, other methods of control often rely on tables of fixed time settings and an appropriate table is selected according to traffic conditions.

Simulation modeling often requires large amounts of computer time. Also, finding the optimal signal setting for even a fixed time control system is a formidable mathematical and computational problem. The problem is further complicated by the stochastic nature of the responses. Of the many techniques that are available for optimizing a stochastic function of several variables, a search technique based on simplex Evolutionary Operation is simple and efficient (4). This automatic search technique can rapidly approach and attain optimum operating conditions. The calculations and decisions are so formalized that they could be executed automatically by a digital computer.

Objective

The objective of this research is to employ a search technique based on Evolutionary Operation for improving the settings of a network of traffic signals in order to maximize the performance of the overall traffic flow. This would be accomplished by utilizing a stochastic traffic network simulator to generate responses corresponding to traffic signal settings which the search procedure would determine. Pertinent factors which must be considered are:

1. Physical dimensions of the network model.
2. Physical characteristics of the network model.
3. Average vehicle speeds.
4. Vehicle arrival rates for every entry point of the system.
5. Number of signalized junctions tested.
6. Starting cycle length (the time period required for one complete sequence of signal indications (1)).

7. Starting signal splits or phases (a part of the cycle length allocated to any traffic movement receiving the right-of-way (1)).
8. Percentages of through and turning traffic at each junction in the model.
9. Starting signal offsets (the number of seconds or per cent of the cycle length that the green indication appears at a given traffic control signal after a certain instant used as a time reference base (1)).

Also, it is the intent of this investigation to attempt to formulate a basis of comparison between the effectiveness of existing methodology and the proposed optimization scheme utilizing Simplex Evolutionary Operation. In particular, a comparison with the method of steepest ascent will be made.

Assumptions

In order to simplify the search process and reduce simulation time certain assumptions are made. Vehicles operating in the system are considered to be of uniform size and have the same performance characteristics. An average constant speed of the vehicles is used throughout any street segment in the network which naturally prohibits overtaking and passing. The physical dimensions of the network primarily consist of street distances between intersections, street widths and the number of lanes in each street segment. The distances between intersections are measured from the geometric centers of the intersections. Therefore, the time consumed by a vehicle passing through an intersection is assumed to be zero.

Amber light settings are considered as being incorporated into the length of green indication. Green arrow and pedestrian indications are not taken into account. However, pedestrian traffic should be considered as an important factor in any street network. Therefore, for purposes of this research, green light indications should be long enough to accommodate the minimum safe pedestrian crossing time at any intersection or a practical vehicle operational minimum time whichever is greater (1).

Vehicle generation and termination take place only at the boundaries of the network. The vehicle arrival rates into the system are assumed to follow a Poisson process. There is no provision for the generation or termination of vehicles within the network such as parking facilities. Vehicle movements at any given intersection are randomly executed with respect to constant movement probabilities derived from vehicle count data. Accidents are not considered in the operation of the simulation.

Vehicle acceleration and deceleration between stops and the average vehicle speed are assumed to be instantaneous. A delay time is used to account for driver reaction in response to the receipt of a green indication. Left turning lanes are assumed to be incorporated into existing primary traffic lanes. A vehicle waiting to execute a left turn will be allowed to "sneak" through the intersection if the signal indication changes from green to red. Further, a vehicle is permitted to "sneak" through opposing traffic on a left turn or from a stop sign if a safe interval is detected.

Definition of Terms

The following is a list of terms extracted from the Traffic Engineering Handbook (1) that are commonly used throughout this thesis.

Time-Cycle--The number of seconds required for one complete sequence of signal indications.

Cycle Split or Interval--Any one of the several divisions of the time cycle during which signal indications do not change.

Color Sequence--A predetermined consecutive order of appearance of signal color indications during successive intervals within a total time cycle.

Traffic Phase--(a traffic movement) a part of the time cycle allocated to any combination of traffic movements receiving right-of-way simultaneously during one or more intervals.

Offset--The time difference or interval in seconds between the start of the green indication at one intersection as related to the start of the green interval at another intersection along the same street, or in the same signal system.

Fixed Time or Pretimed Controllers--

- a. Operate on pretimed timing sequence for cycle length, split and offset (if part of a system).
- b. Timing is based on observed conditions.
- c. May have one timing program which operates at all times, or
- d. May have several timing programs for the various periods of daily vehicular traffic.

Traffic Actuated Controller--

a. Operates on a timing sequence which is controlled by traffic (variable cycle length, splits and offsets).

b. Traffic actuation may be initiated by pedestrian push button, vehicle detectors on minor streets, or detectors on both main and minor streets.

Through Band--The time in seconds elapsed between the passing of the first and last possible vehicle in a group (platoon) of vehicles moving in accordance with the designed speed of a progressive signal system.

Progression--A time relationship between adjacent signals permitting operation of groups of vehicles at a planned rate of speed.

Signal System--Two or more individual signal installations having a fixed time relationship to each other.

System Evaluation

In order to properly evaluate a set of signal settings, some measure of effectiveness is necessary. The most commonly mentioned performance criterion indicated by a literature survey is some form of vehicle trip delay. However, intuitively speaking, delay may not be the "best" or at least the only criterion that should be considered. The behavior of the driver plays an important role in traffic movement. He may possibly tolerate some added trip delay in order to decrease the irritation caused by multiple stops. Another important factor that may be considered is the cost of vehicle operation in urban traffic. A driver more than likely would enjoy a more economical operation of his vehicle as a result of an efficiently controlled street network. In

this investigation, the measure of effectiveness would be minimizing the average delay per vehicle with a trade off of average stops per vehicle. In addition, both criterion may be costed over a particular period of time and a possible total savings in vehicle operation determined.

CHAPTER II

LITERATURE SURVEY

The automobile oriented American is reflected in the phenomenal growth of passenger car use. From 1940 to 1960 passenger car usage increased 221 per cent in urban areas alone. Although some improvements in urban street networks have been made, there has been a relatively small increase in the amount of roadway in which this steadily increasing volume of automobile traffic is permitted to move. Between 1930 and 1960, urban population increased from 56 per cent of the total population to 60 per cent. Estimates indicate that it will reach 75 per cent of the total population by 1980 (1). With this it is also expected that there will be a substantial expansion of automobile usage in the urban areas.

One suggested means of dealing with the resulting traffic jams and congestion is to improve the utilization of the current road system (2) and (3). Urban areas are, in general, characterized by their geometric grid network and associated heavy traffic volumes. Because of this there is a high concentration of traffic signals employed at street network intersections. Here, the traffic signals are used as control devices to alternately assign the right-of-way to orthogonally opposed traffic flows. It is to these intersections that approximately 80 to 90 per cent of all vehicular delay can be attributed (5). Without changing the physical characteristics of the network, the efficient

operation of these signals then become a major factor in the reduction of avoidable delay. The values of the control parameters selected (traffic light cycle length, red and green light time split, etc.) are, at times, critical if service to motorists is to be optimized (6).

By utilizing simulation, traffic control systems can be evaluated on a computer instead of a street. As simulated automobiles pass through a street network, their stops and delays are easily recorded for evaluation purposes (2).

A traffic simulation model has been developed by Schwartz (3) for the study of traffic flow in a signal controlled network. It was applied to the Back Bay area of Boston. Various alterations in the signal patterns were coded into the model, tested, and the desirability of changes in the existing signal pattern was shown. This simulation model provides a convenient tool for the testing of various innovations for the improvement of traffic signal systems. This article is complete with flow charts and a computer program listing which would prove invaluable in the development of a general traffic network model.

A similar traffic network model was described by Wagner and Gerlough (7). Their simulation also provided a foundation for the development of practical tools for use in the solution of traffic operations and control on city streets.

A data-sampled system for the computer control of traffic in urban networks is described by Miller (5). He utilized a digital computer which inputs data from detectors placed at the approaches and stop lines for every junction. The primary form of control described in the article is that of signal timing control.

The National Cooperative Highway Research Program (6) conducted a study of a computer controlled traffic signal system for an urban street complex. The approach taken was to synthesize this type of signal system for a small city such as White Plains, New York. A control doctrine of minimum delay was adopted and a generalized expression was developed to evaluate delay for a subnetwork of intersections. This research also presents an idea of the cost involved in implementing a digital computer control system for medium and small cities.

A digital computer controlled traffic signal system is in use in Wichita Falls, Texas. This traffic responsive system controls 77 intersections of which 55 are located in a 47 square-block downtown area. The value of this traffic control system is exemplified by a recent evaluation which indicated a reduction of stops by 16.3 per cent, average vehicle delays by 31.1 per cent and accidents by 8.5 per cent (8). There is an indication of savings to motorists in operating costs as a result of the reduction in stops and delays.

The city of San Jose, California, is also utilizing a digital computer to control a 59 traffic signal system (9). This system uses fixed-time progressions determined by the time of day. This type of system adapts somewhat to general traffic conditions by introducing several different patterns of splits and offsets to be used for such conditions as light, inbound, outbound, and average flow conditions.

La Batt of Automatic Signal (10) has proposed a computer controlled traffic responsive signal control system for Charleston, South Carolina. The master controller proposed operates on traffic performance data (speed and volume) for determining signal system cycle

length, splits, and offsets. The basis of the timing program is an analog version of the time-space diagram--the most elementary traffic engineering method of relating adjacent intersections to each other.

Computer controlled traffic systems permit real time control on both an individual intersection and area wide basis, i.e., vehicle detectors in the streets provide information with which to set the signals more or less immediately or continuously. Miller and Little (2) suggest that work on fixed-time systems would provide a more fruitful base for research. Their reasons are twofold. First, the geographic size, traffic conditions, and/or monetary status of most cities will not permit the installation of sophisticated signal control systems. Second, computer controlled systems presently work to a considerable extent from tables of fixed-time settings, an appropriate table being selected by the computer according to traffic conditions.

To maintain a fixed-time system, the total cycle length at all installations normally must be equal (1). In unusual cases one installation might operate at double or half the cycle length of the system.

Simulation frequently consumes large amounts of computation time. Further, finding the exactly optimal fixed-time setting for a complex model of a network of signals is a formidable mathematical and computational problem. The decision variables are numerous--each signal has a given split and an offset and the network as a whole has a cycle length. Miller and Little (2) indicate that there is no exact solution to the problem, but all sensible approaches should be examined. They suggest finding starting settings that would be expected to be good, evaluate them by simulation and then apply some systematic improvement procedure.

One interesting improvement technique suggested by Rangarajan and Oliver (11) is the determination of an optimal allocation of servicing (green light) periods to a facility-serving N incoming traffic streams. The procedure used in this article is very similar to that of the classical inventory economic lot size model with no back orders allowed. The writers also develop an algorithm to determine optimal cycle and fractional service times. This study is basically limited to isolated signalized intersections.

A study was made to ascertain whether traffic intersections can be effectively coupled on the basis of traffic behavior (12). Information on vehicular platoons was collected at four sites in London, England, and was analyzed with particular emphasis on the phasing or synchronization of neighboring signalized intersections for minimum delay. This data gives a measure of how platoons diffuse as they move from one intersection to the next. The analysis indicates that the diffusion process can be taken into account in the setting of the signals. The optimal offset time that would minimize delay is shown to be a linear function of the distance from the issuing signal.

A traffic control strategy for urban centers subject to peak hour traffic jams is presented in an article by Longley (13). He suggested that each controlled junction should adjust the given time split on the basis of queue length ratios on its various arms. He further states that such a control system will respond to changes in traffic flow rates and to factors affecting saturation flow rates across the junction, e.g., breakdowns, badly parked cars, etc.

Little (14) presents a mixed integer formulation for synchronizing

traffic signals which can be extended to networks. An objective function is formed from the band widths of the arteries. This is one of the first attempts, particularly in the case of street networks, to develop some sort of optimizing format for synchronizing traffic signals. Branch and bound methods are given to solve the network problem by section.

Research conducted by Robertson (15) utilizes a hill climbing technique for the improvement of traffic signal settings. Basically the scheme used improves only offsets and phasing. The optimizing process used does not include the cycle time. An optimum cycle time for the network is assumed.

There are several efficient search procedures which could be utilized for optimizing a function of several variables, as may be found in the traffic signal setting problem. Among these search procedures are Powell's Method (16), Fletcher and Powell's Method (17), Hooke and Jeeves' Pattern Search (18), and Simplex Evolutionary Operation or Sequential Simplex Pattern Search (4), (19), (20), (21), (22), and (23). Of these, a search procedure based on Simplex Evolutionary Operation is probably the simplest and most efficient.

In 1962, Spindley, Hext and Himsworth (4) introduced an idea for tracking optimum operating conditions by evaluating the output from a system at a set of points which form a simplex in the factor space. The procedure continually forms new simplices by reflecting one point in the hyperplane of the remaining points. The idea is clearly applicable to the problem of optimizing a mathematical function of several variables. This technique when utilized for empirical optimization, is an outgrowth of Box's rudimentary "steepest ascent" procedure of Evolutionary

Operation (abbreviated to EVOP) (19) and (24).

Simplex EVOP is an automatic procedure which can rapidly approach and attain optimum conditions. Its calculations and decisions are formalized and can be executed automatically by a digital computer (4). Another important feature of simplex EVOP is that if the form of the objective function is unknown, only the levels of the controllable variables and responses corresponding to these variables are necessary. A discussion of a modified simplex EVOP for the minimization of a function of N variables is presented by Nelder and Mead (23). The method modifies the usual simplex procedure according to the local landscape; elongating down long inclined planes, changing direction on encountering a valley at an angle, and contracting in the neighborhood of a minimum.

The versatility of simplex EVOP may be exemplified by its successful utilization in industry for several different processes. Three such examples are in production planning and inventory control (25), process improvement (21) and machine center capacities control in a job shop (22). Using simplex EVOP in conjunction with a traffic simulation model, it would therefore seem possible to determine an optimal set of traffic signal settings for improving the quality of traffic flows.

CHAPTER III

A TECHNIQUE FOR IMPROVING SIGNAL SETTINGS

This research utilizes a search technique based on Evolutionary Operation for determining traffic signal settings and evaluating their effectiveness based upon responses derived from a digital simulation. A discussion on the search technique and its employment in conjunction with a traffic network simulator is contained within this chapter.

The Simplex Design

Simplex Evolutionary Operation and various modifications to the basic technique have already been noted in the literature survey. Outside of generalizing the basic construction of the simplex design, the particular method proposed in the original article by Spindley, Hext, and Himsworth (4) and later by Glenn (20) is the one to be utilized in this investigation.

To use simplex EVOP for optimization of a function of K variables, it is necessary to construct a simplex. The discussion of simplex EVOP in this section is written for the minimization of a function. Should maximization be required, replace the word "lowest" with "highest." A simplex is an orthogonal first order experimental design which requires only one more experimental point than the number of variables under consideration (20). Therefore, for K variables, there are $N = K + 1$ design points (rows of the design matrix). The design is formed by a regular sided figure with $N = K + 1$ vertices so situated in the K

dimensional space that the cosine of the angle formed by any two vertices with the center of the design is equal to $-1/K$. For example, if $K = 2$, the design would be an equilateral triangle, and for $K = 3$, the design is a tetrahedron.

A slight variation of the standard simplex design (4) for K variables is used. A regular simplex is then specified by the following $(K + 1) \times K$ design matrix:

$$D_o = L \begin{bmatrix} 0 & 0 & 0 & \dots & 0 \\ p & q & q & \dots & q \\ q & p & q & \dots & q \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ q & q & q & \dots & p \end{bmatrix} + \begin{bmatrix} S_1 & S_2 & \dots & S_k \\ S_1 & S_2 & \dots & S_k \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ S_1 & S_2 & \dots & S_k \end{bmatrix}$$

$$\text{where } p = \frac{1}{K\sqrt{2}} \quad (K-1 + \sqrt{K+1})$$

$$q = \frac{1}{K\sqrt{2}} \quad (\sqrt{K+1} - 1)$$

L = The desired length of a side.

S_i , $(i = 1, 2, \dots, K)$ = coordinates of the chosen origin.

The rows of D_o are the coordinates of the design points.

The Simplex EVOP Procedure

Values of the objective function, or response, are determined for each set of coordinates in the design matrix. By comparing the values of the objective function of these $(K + 1)$ points, the vertex or point with the highest response or the worst point in minimization is selected for replacement. The direction of minimum response out of the simplex would then proceed in some direction from the center of the design out of the side opposite to the highest response. The least desirable point is then discarded and is replaced by a point with a lower value to form a new simplex of $(K + 1)$ points. This process is repeated until the point corresponding to the minimum value of the objective function is achieved.

An example of this procedure for two variables is shown in Figure 1.

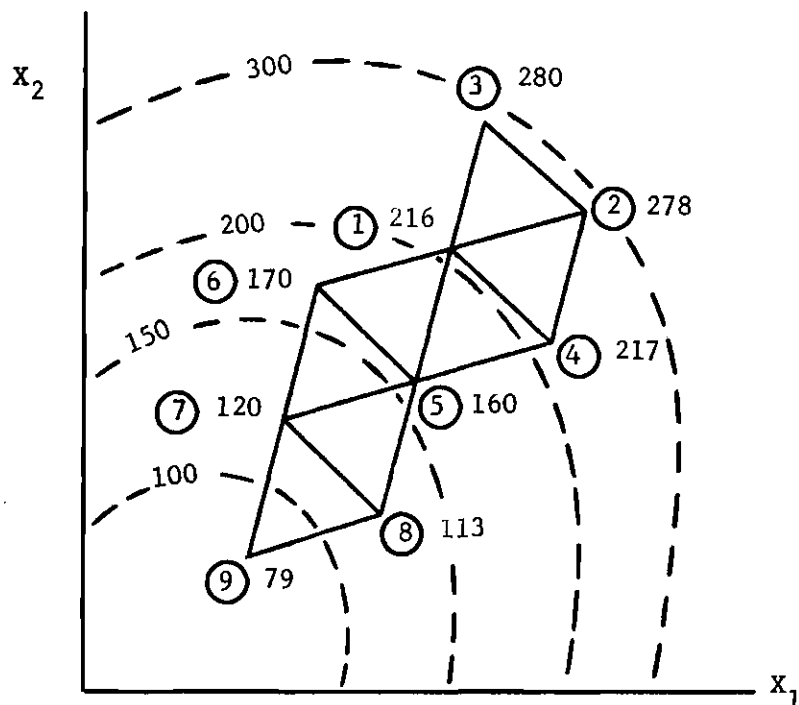


Figure 1. Example of a Two Variable Simplex EVOP.

In this figure the numbers of the experimental points are circled and the response is given for each.

For the formation of a new simplex, the coordinates at the new experimental point are determined in the following manner:

Let the rows at the current simplex S_0 be denoted by the vector \underline{d}'_i , for $i = 1, \dots, K + 1$. If the least desirable response (i.e., the maximum observation) occurred at \underline{d}'_i , the determination of the coordinates of the new response is given by:

$$\underline{d}'_i^* = \frac{2}{K} (\underline{d}'_1 + \underline{d}'_2 + \dots + \underline{d}'_{i-1} + \underline{d}'_{i+1} + \dots + \underline{d}'_{K+1}) - \underline{d}'_i$$

The coordinate of the new point is twice the average of the coordinates of the common points minus the coordinates of the rejected point. In other words the new point is merely the mirror image of the discarded coordinate in the common face of the simplex.

Therefore, the simplex EVOP procedure of a minimization is given by the application of the following rules (4).

Rule 1: Ascertain the highest response Y_i of Y_1, Y_2, \dots, Y_{K+1} . Complete a new simplex by excluding the point \underline{d}'_i corresponding to Y_i and replacing it with \underline{d}'_i^* .

Since the responses are subject to error, there is a possibility that the system of simplices may become anchored to a spuriously low response. To reduce the risk of this occurring, Rule 2 is applied.

Rule 2: If a result has occurred in $K+1$ successive simplices and not then eliminated by Rule 1, discard this result and replace it by a new observation at the same point.

The reason for this rule is that if the point is at a true optimum, the replication will again be clustered about it. If it were low due to error, the replication will probably not be so low and would be eliminated in due course.

When the responses are not subject to error (i.e., when the procedure is used for numerical optimization), the maximum age (maximum number of successive simplices) is defined to be that which can be attained by a point in normal progression. Only when this age is exceeded could it be concluded that the region of the optimum had been reached. Table 1 provides maximum ages for various K.

Table 1. Maximum Age Attainable by a Point.

K	Age	K	Age	K	Age
2	3	6	11	10	21
3	5	7	13	15	36
4	7	8	16	20	52
5	9	9	18	30	92

Over the range the maximum age can be approximated by the fitted equation

$$\text{Age} = 1.65K + 0.05K^2$$

where K = number of variables.

A spuriously high response will generally be eliminated quickly but may cause some oscillation from one simplex to the previous one. To prevent this from occurring Rule 3 is applied.

Rule 3: If Y_i is the highest reading in the simplex, S_0 , and Y_i^* is the highest reading in the new simplex, S_1 , do not move back to S_0 . Instead reject the second highest reading of S_1 .

The application of these rules causes the system of simplices to circle continuously about the optimum rather than oscillate over a limited range. It also makes progress possible if, by chance, the system of simplices should straddle a "ridge" in the factor space.

The three rules given above may be summarized by the following:

Move by rejecting the highest response unless (a) another response is too old, in which case the latter is removed, or (b) such a move would cause a return to the previous simplex in which case the next favorable direction is tried.

If there are constraints on the levels of the variables of the form $a \leq x \leq b$ and simplex $p-1$ generates a point in the p^{th} simplex that would violate the constraint, the second most favorable direction from $p-1$ would be used.

Employment of Simplex EVOP

In this research, simplex EVOP utilizes a traffic network computer simulation as an integral part of the optimization procedure. The sequential simplex technique as it is sometimes called is illustrated in Figure 2. The routine commences by entering with a set of K signal

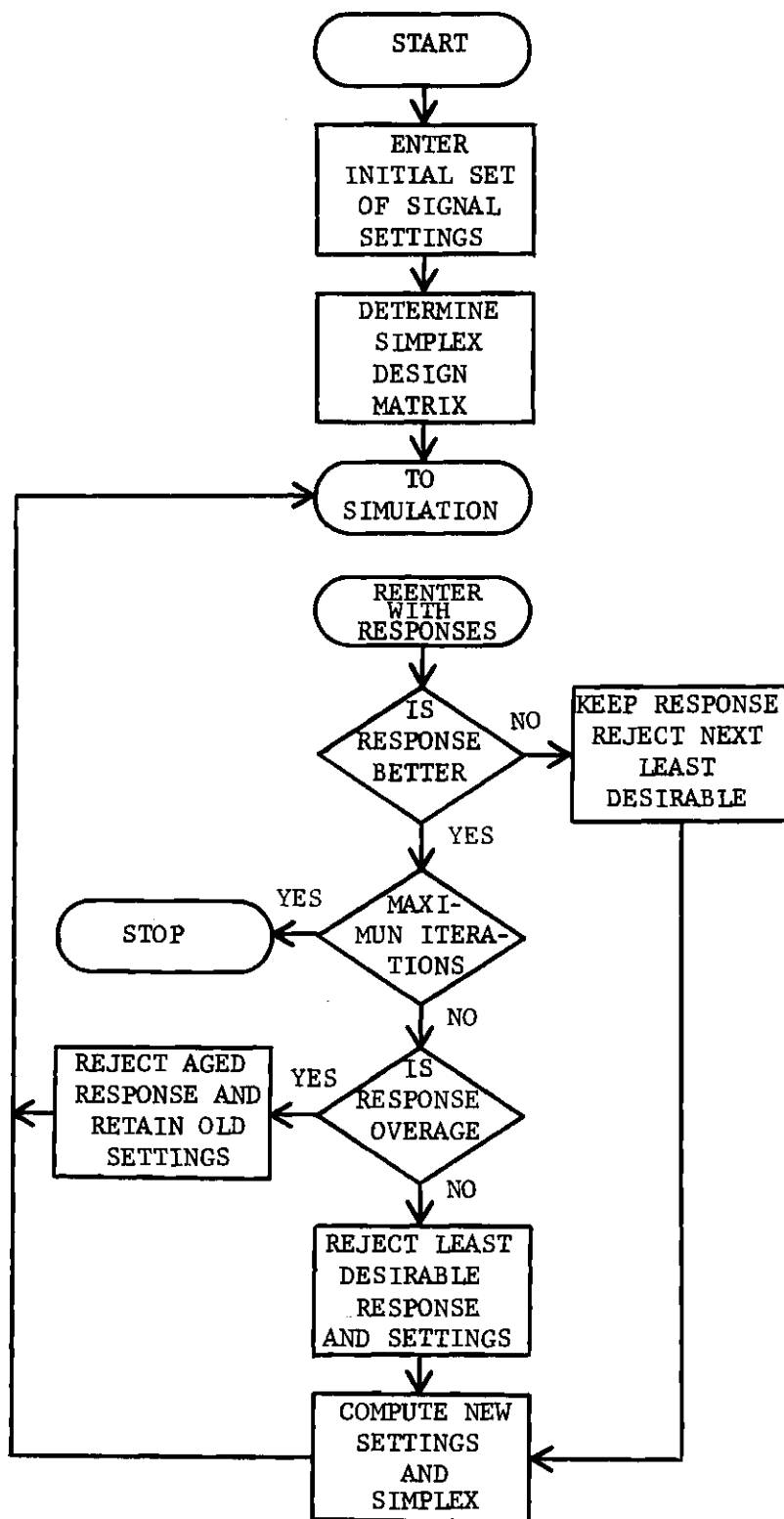


Figure 2. Sequential Simplex Procedure Flow Chart.

setting variables which are expected to be reasonably good. A $K+1 \times K$ design matrix is automatically formed where there are $K+1$ sets of signal settings (or points). A simulation is conducted for each point and a corresponding response is recorded. The least desirable response is rejected and a new simplex is formed by reflecting the point corresponding to this response in the hyperplane of the remaining points. This process is iterative and continues until an optimal set of signal settings is attained. This occurs when the system of simplices is caused to circle continuously about the optimum response. A computer listing of this routine may be found in Appendix A.

The Simulation Model

A number of urban street network models are presently available for the study of traffic flow. One such model developed by Schwartz (3) appears to have the majority of necessary characteristics essential to the evolution of signal settings. His model was programmed in the General Purpose Systems Simulator II (GPSS II) computer language. The flow diagrams and charts in Schwartz's report were useful in coding a slightly modified model in the more general FORTRAN IV computer language to better accommodate the mathematical processes of the search technique used in this research. This is a more realistic model than others examined in that vehicle generations and movements are based upon stochastic processes. A macro flow chart of the traffic network system is illustrated in Figure 3. Before a simulation is performed, the model is first initialized with data pertaining to the network. During this period a simulation is conducted to "load" the network with vehicles.

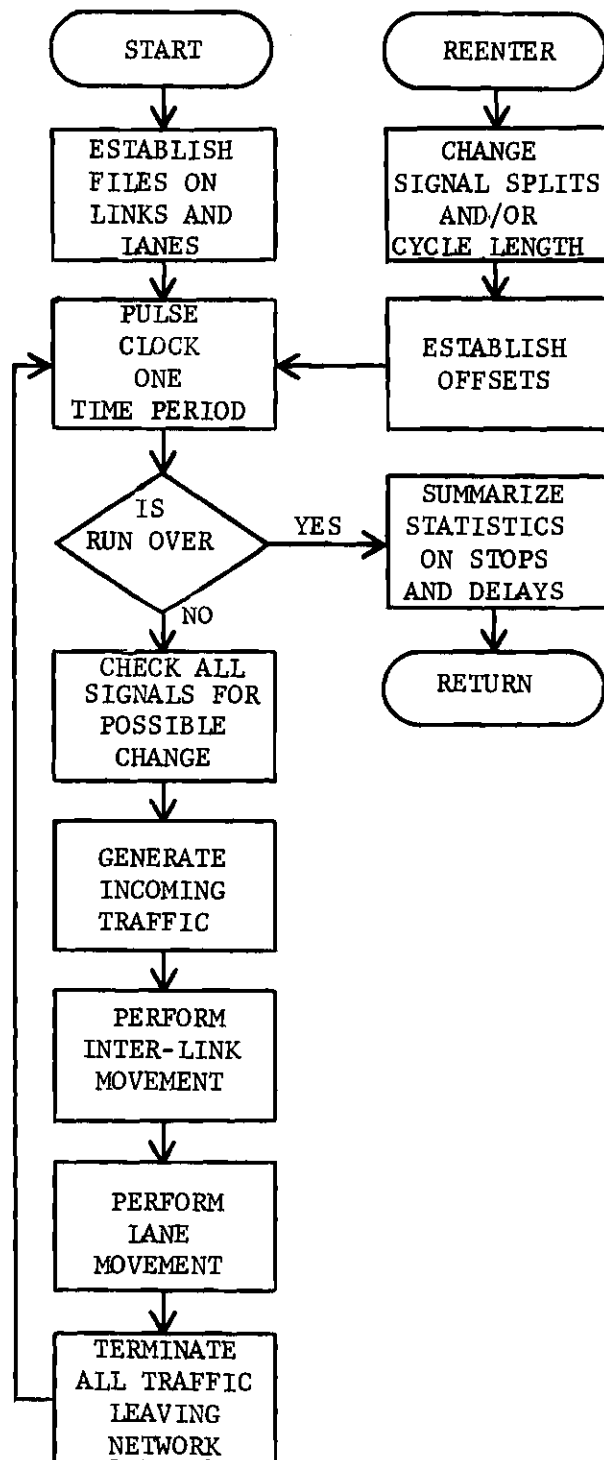


Figure 3. Traffic Simulation Network Flow Chart.

On call from the simplex EVOP procedure, the simulation is activated utilizing signal settings generated by the current design matrix. The incoming traffic generator has vehicles enter the network from boundary points. The vehicles are then moved on streets to intersections. Where an intersection is encountered, crossing and turning movements take place in accordance with traffic signals. Once vehicles enter a new street link, they are assigned to appropriate lanes. Traffic reaching the boundary points is then terminated. Upon completion of a specified run time, the output data is summarized and sent to the simplex EVOP procedure for evaluation. The computer listing of this simulator and sample data are presented in Appendix B.

CHAPTER IV

THE CONDUCT OF THE EXPERIMENT

Prior to the execution of the experiment itself it is necessary that several analytical and emperical parameters be defined. This chapter discusses these parameters and the intended sequence of the experiment.

Discussion

The sequential simplex technique in this research is being proposed as a method for providing an improved set of signal settings for a traffic network. The sequential simplex search routine is used as the main program and the traffic simulator as a subprogram. Before conducting the actual experiment, a number of decisions are made with regard to the network to be tested, the decision variables, the objective function, the length of a side of the simplex and the span of simulation time.

The Network

The traffic network analyzed is a six intersection portion of the central business district of Wichita Falls, Texas. As related in the literature survey, the Wichita Falls traffic signal system is computer controlled. According to traffic conditions, the computer selects the "best" set of signal settings from tables of fixed time settings. Observed traffic counts for three one-hour periods of the day - the A.M. peak, an off-peak and the P.M. peak - are used for the traffic network

simulation. These three sets of counts provide for realistic average vehicle arrival intervals (in seconds) into the network (see Table 2) and for fixed percentages of vehicle through and turning movements at each intersection (see Figures 4, 5, and 6). The one-way streets each have three lanes. Scott has four lanes. Lamar has three lanes which alternate midblock to accommodate turning movements. Lamar is therefore considered to have four lanes in accordance with the assumptions for turning lanes as outlined in Chapter I. The computer selected signal settings in effect at the time of each traffic count are used as "good" starting signal data to be improved by the sequential simplex technique (see Table 3).

The posted speed limit of 30 miles per hour is the average vehicle speed used by the vehicle simulator. This coincides adequately with a simulated clock increment of one-half second.

The main thoroughfares in the network are designated Scott and Lamar. The remaining streets perpendicular to Scott and Lamar are considered secondary thoroughfares. It is along the main thoroughfares where the signal settings are sought to be improved. A reduction in the number of variables to be tested at each intersection results. For example, a percentage of the cycle time designated as the green period on the main thoroughfare would merely have to be subtracted from 100 per cent to attain the green interval on the secondary thoroughfares.

Objective Function Formulation

As indicated in Chapter I, this research utilizes as a measure of effectiveness the minimum average delay time per vehicle and average stops per vehicle. Initially, the objective function to be employed

Table 2. Vehicle Arrival Intervals For Periods Tested.

Period	Node	Interval (seconds)
AM	101	10.753
	102	29.412
	104	25.641
	105	17.544
	107	8.621
	109	17.241
	110	13.158
Off-peak	101	6.849
	102	9.259
	104	10.204
	105	20.833
	107	6.173
	109	12.500
	110	9.709
PM	101	6.622
	102	7.353
	104	7.576
	105	20.408
	107	6.803
	109	10.638
	110	8.547

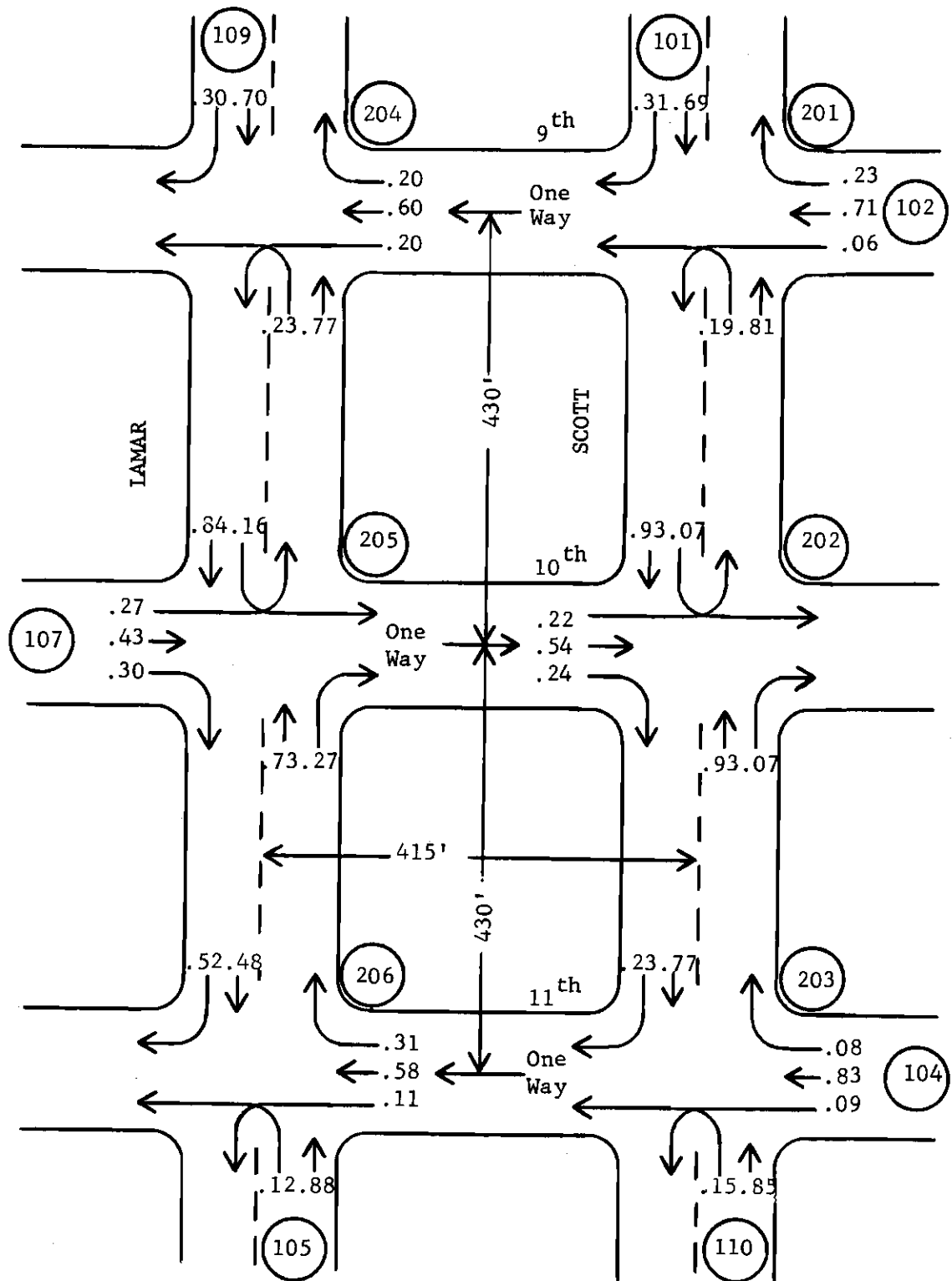


Figure 4. Network Layout for A.M. Peak (7:30-8:30)
(Not to Scale).

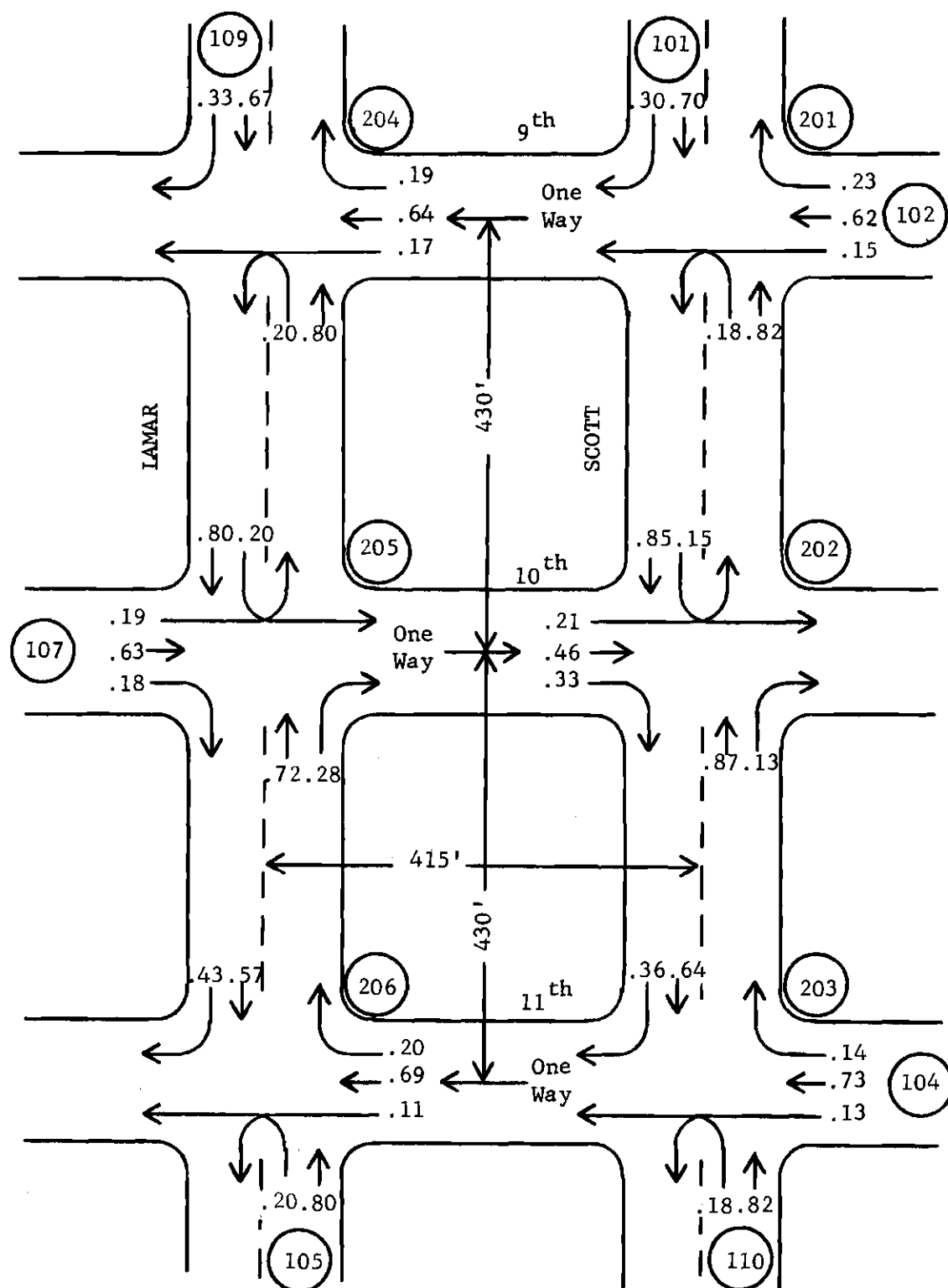


Figure 5. Network Layout Off-peak (3:30-4:30)
(Not to Scale).

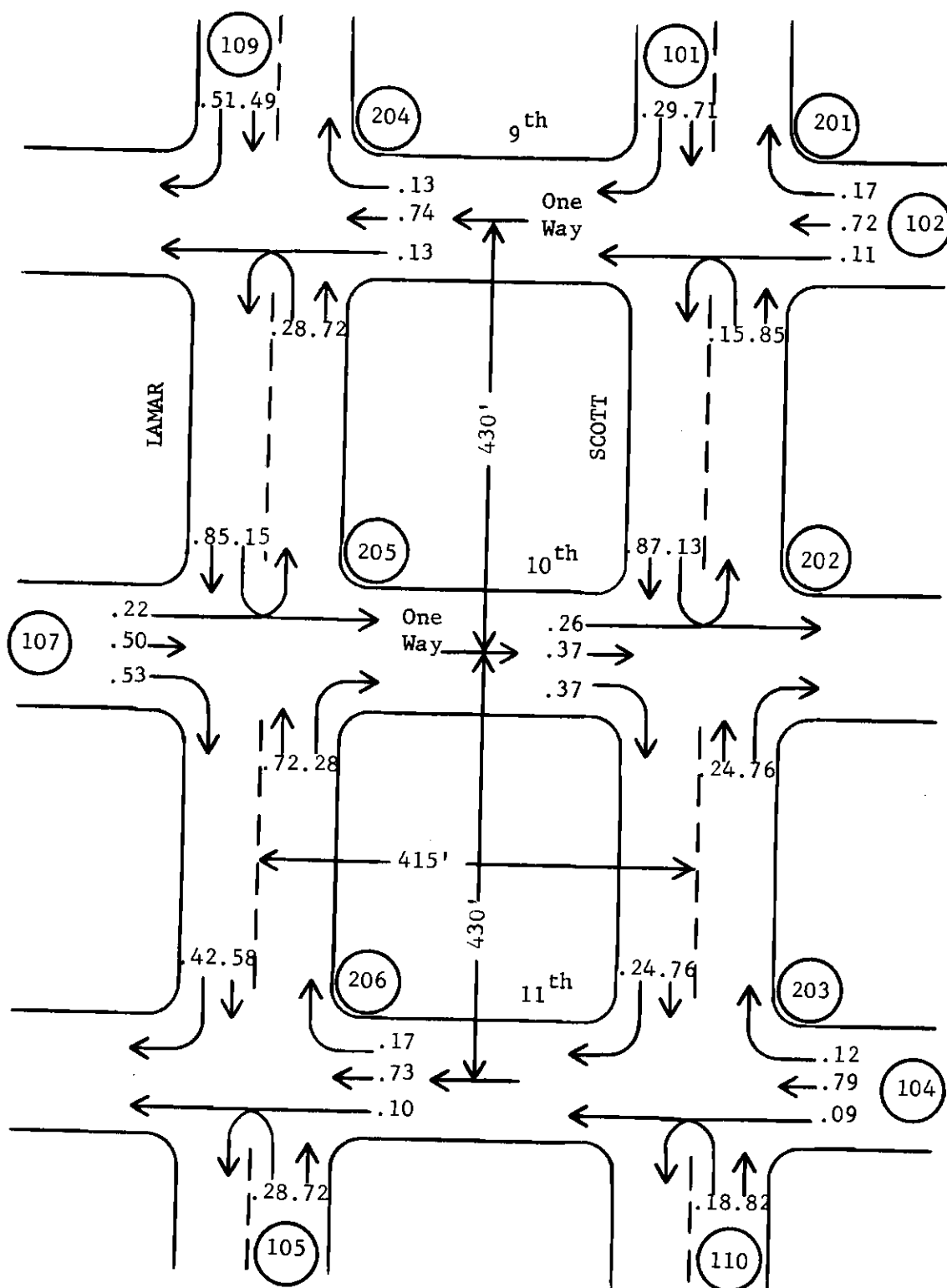


Figure 6. Network Layout P.M. Peak (5:00-6:00)
(Not to Scale.)

Table 3. Starting Signal Settings For Points Tested.

Period	Intersection	Cycle Time (sec.)	Offset as percentage of cycle time	Green Phases on main thoroughfares of Scott and Lamar as a percentage of cycle time
AM	201	49	24	51
	202		74	51
	203		74	51
	204		49	51
	205		49	51
	206		0	51
Off-peak	201	53	24	51
	202		74	51
	203		74	51
	204		51	51
	205		51	51
	206		0	51
PM	201	53	24	51
	202		74	51
	203		74	51
	204		49	51
	205		53	51
	206		0	51

was the minimizing of total vehicle delay and total vehicle stops. However, some initial experimentation indicated a better objective would be to minimize average stops and delay per vehicle.

Many references utilize "average" measures of effectiveness for traffic network evaluation, however none actually define them precisely. Therefore, for purposes of this research, average stops and delay in seconds per vehicle are defined as follows: For any simulation run totals of vehicle stops and seconds of delay are accrued. Total delay is defined to be the time waiting for a green indication plus any starting delay time. These two values are then divided by the total number of vehicles contributing to the delays and stops to determine the average of each during a particular simulation run. The contributing total of vehicles is further defined as the number of vehicles in the system at the commencement of the simulation plus those that enter during the conduct of the simulation.

The minimizing of the objective function of either average delay or average stops per vehicle provides for more stable responses during replication. However, when delay is reduced, the driver spends less time on the average waiting at each red indication, but the average number of stops increases considerably. On the other hand, a reduction in the number of stops permits the driver to travel further during a green indication without stopping, but once stopped has waiting time period increases due to longer red indications.

At this point two possible alternatives are available. One is to minimize delay subject to a constraint on stops or vice versa. The other, and perhaps more reasonable, alternative is to develop an

objective function consisting of both average delay and stops per vehicle. This latter alternative is the measure of effectiveness used in this research.

In order to combine the two values into a single objective function, the trade-off of one stop being equated to 14.5 seconds of delay is used. This value was derived by using figures published by the American Association of State Highway Officials (8). This study indicates that it costs a motorist \$0.00051 per second of delay and \$0.0074 per stop. The delay figure includes a figure of eight thousandths of a cent per second of engine idle time and 43 thousandths of a cent for each second of delay the driver encounters. Assuming that these values are correct by today's standards, one hour of delay to the motorist would be worth \$1.55.

If the cost of a stop is divided by the cost of a second of delay the following relationship is derived:

$$\frac{\$0.0074 \text{ per stop}}{\$0.00051 \text{ per second delay}} \approx 14.5 \text{ seconds delay per stop}$$

The objective function is then defined as: $Z = \bar{D} + 14.5\bar{s}$

where \bar{D} = the average delay per vehicle in excess of unimpeded movement through controlled intersections* of the network.

* A controlled intersection is one which utilizes either traffic signals or stop signs to regulate the assignments of rights-of-way to approaching traffic.

\bar{s} = the average number of stops per vehicle caused by the controlled intersections* within the network.

This corresponds closely to that used by Miller and Little (2) for improving signal settings using various techniques they describe. However these authors make no attempt to derive the constant used to convert stops into seconds of delay.

This research then attempts to seek a local minimum of the above objective function. Minimizing this objective function intuitively minimizes motorists' costs for delay and stops while traversing a traffic network. The cost per vehicle can be obtained by multiplying the objective function by \$0.00051.

Due to the nature of the sequential simplex routine and limited computer time, the following conditions are adopted for the determination of improved traffic signal settings:

(i) The signal settings derived should be both practical and realistic. First, the variables (the signal settings) naturally cannot be of negative value. This then introduces a nonnegativity constraint. Secondly, the minimum green interval for each phase is set either by pedestrian requirements or by a vehicle operational minimum of 12 seconds (1). Using a minimum pedestrian starting time of five seconds and a walking speed of four feet per second, the minimum pedestrian time is determined by: $G = 5 + \frac{W}{4}$

* A controlled intersection is one which utilizes either traffic signals or stop signs to regulate the assignments of rights-of-way to approaching traffic.

where G = the minimum green interval in seconds.

W = the width of the street being crossed by pedestrians during the green phase.

From an inspection of dimensions of the selected network, all streets have a width of 50 feet except Scott which is 60 feet wide. Therefore a minimum green interval across Scott should be 20 seconds and 18 seconds for all others. For simplicity, a constraint of a minimum green interval of 18 seconds is imposed for the entire network. For purposes of this thesis a two-second decrease in the minimum green interval across Scott is considered insignificant. Further, in such situations where a minimum green interval would not permit pedestrians to cross a wide street, a median could be installed as a pedestrian refuge island.

A light that remains on red in excess of 120 seconds may cause drivers to consider that the signal is malfunctioning and will tend to "run" through the red indication. This third factor is monitored to insure that it is not violated and not employed as a constraint in order to keep computer running time to a minimum.

(ii) The set of signal settings should produce a minimum response for at least two replications and be equal to another minimum response for a third replication over all sets which remain in the category described in (i) above. An exception to this would be a set of signal settings which produces still a better response but cannot be replicated due to the constraints or maximum computer time prohibiting the progression of the simplices.

The Controllable Variables

The controllable variables are of course the signal settings themselves. There are three sets of variables to be improved: the green splits or phases, the offsets and the time cycle length. The green splits and the offsets are entered in the simplex matrix as percentages of the cycle length. A scaling convention is established making all variables integer values.

The Simplex Side Length

The length of the side of a regular simplex plays an important part in attaining improved coordinates and associated responses. The side length is dependent upon the number and types of variables in the simplex design. As a result, every starting matrix would require the side length to be empirically adjusted in order that the best possible stepping and responses may be obtained. This then requires several computer runs for each period and variable combination tested.

The Simulation Time

Of the references examined concerning urban street network simulation only one indicates any specific simulated street time which could provide adequate information for network analysis. Miller and Little (2) used a 575 second street simulation of which 50 seconds of that time is ignored in order to insure that the network is filled with cars. This research utilizes a 600 second simulation for each set of signals tested. In addition, one simulation is initially used to load the system before any signal settings are tested at all.

The Experiment

This experiment is concerned with the testing and improving of signal settings which were in use in Wichita Falls at the times that the traffic counts were taken. For each of the three periods tested, the controllable variables are increased in number by type. For example, the initial signal settings selected to be improved are the six offsets for each period. These were selected because efficient vehicle progression relies heavily upon the proper sequencing of the traffic signals. These are problems one, two, and three. Next, in problems four, five, and six, the controllable variables assume inclusion of the offsets and cycle time. This choice was made because the correct cycle time, when suited to the needs of traffic for particular periods of the day and employed in conjunction with other improved variables, greatly enhances the quality of traffic flow. Further, the above variable combination seeks to improve cycle time simultaneously with different variables, whereas other signal setting improvement schemes select the cycle time independently. By similar reasoning, problems seven through 15 are developed. All problems investigated are described in Table 4.

Table 4. The Sets of Controllable Variables
to Be Tested for Each Problem.

Problem			Controllable Variables (Number)
AM	Off-peak	PM	
1	2	3	Offsets (6).
4	5	6	Offsets (6) and Cycle Time (1).
7	8	9	Green Splits (6) and Offsets (6).
10	11	12	Green Splits (6), Offsets (6) and Cycle Times (1)
13	14	15	Green Splits (6), Offsets (6) and Cycle Times (6)

CHAPTER V

DISCUSSION OF RESULTS

Improvement of traffic signal settings for a street network was successfully achieved through the application of the sequential simplex technique. This chapter presents the methods of attaining significant results, a cost reduction analysis, and a comparison with possible alternate search schemes.

The Results

Simulation runs were conducted for all three periods of the day, each with five different sets of controllable variables. In each problem the initial coordinates of the design matrix, D_0 , were the signal settings in use at the time traffic counts were taken (see Table 2). In all cases tested, the sequential simplex technique provided improved traffic signal settings for the network. The results of each problem studied are presented in Appendix C. The success of this experiment relied heavily upon the attainment of an adjusted simplex side length for each problem and the ensuing verification of the results by the conduct of statistical significance tests.

Adjustment of the Simplex Side Length

The sequential simplex procedure was found to be highly sensitive to the size of the side length employed. Preliminary experimentation was necessary to study what range of values would be appropriate for use in this research.

The side length for every problem design matrix was independently adjusted. In some cases it was deemed necessary to make adjustments to the nearest one-tenth of a unit. The adjustment of the simplex side length was halted when the conditions for the determination of improved traffic signal settings were met.

Verification of the Results

It is desirable to verify that a marked improvement in the traffic signal settings has been achieved. A survey of the results made it readily apparent that the chosen improved signal settings for the A.M. period produced the least reduction in the objective function. The following pooled t test was conducted to ascertain whether or not each set of improved signal settings for the A.M. period would produce a statistically significant reduction in the objective function:

$$H_0: \mu_x \leq \mu_y$$

$$H_1: \mu_x > \mu_y$$

where x = mean objective function value at the initial settings.

y = mean objective function value at the improved settings.

If $t_o > t_{\alpha, n_x + n_y - 2}$ we reject the null hypothesis
where

$$t_o = \frac{\bar{x} - \bar{y}}{\sqrt{\frac{1}{n_x} + \frac{1}{n_y}} \sqrt{\frac{\sum_{i=1}^{n_x} (x_i - \bar{x})^2 + \sum_{i=1}^{n_y} (y_i - \bar{y})^2}{n_x + n_y - 2}}}$$

$$\alpha = 0.01$$

$$n_x = n_y = 25$$

Twenty-five traffic simulations were run utilizing each set of improved coordinates and the initial coordinates for the A. M. period. Appropriate statistical data for each was compiled and the above t test conducted. In all cases a significant improvement in signal settings had been achieved. This is illustrated in Appendix C. For example, in the six variable problem (problem 1), the improvement in the objective function was 3.16 seconds. The results of the t test are tabulated in Table 5.

Intuitively speaking, it may be said that if settings which afforded a small reduction in the objective function were found to be significantly improved statistically, then settings providing larger reductions in responses were also significantly improved. Based on this premise the signal settings for the off-peak and P.M. periods also showed distinct improvement.

Basically, as the number of dimensions of the simplex is increased a corresponding reduction in the response should be detected after the

Table 5. Result of the Pooled t Tests Conducted for the A.M. Period

Problem	t_o
1	3.404*
4	7.023*
7	6.593*
10	6.513*
13	4.240*

*Significant at $\alpha = 0.01$ where $t_{0.01; 48} = 2.407$.

improved settings were determined. In some cases, the results did not bear this out. Larger simplices require a greater number of iterations and consequently more computer time for orientation in the region of optimality. An arbitrarily selected computer run time of 15 minutes was utilized for the experiment. As a result, the larger simplices and perhaps some small ones although already having produced significantly improved signal settings, could not progress toward still better settings because of this computer time constraint.

Vehicle Operating Cost Reduction

A cost effectiveness analysis was conducted to determine the effect of improved settings derived by the sequential simplex technique on reducing vehicle operating costs. No direct cost comparison could be

made with the computerized Wichita Falls traffic control system. However, if certain assumptions are considered valid, an estimated cost savings to motorists can be shown to be accrued by the implementation of the sequential simplex technique. To accomplish this the same parameters used for an evaluation of the Wichita Falls system are adopted, namely a 14 hour day, 90,000 vehicles per day, and a 300 day year (8).

The approach used first assumed that the amount of improvement in objective function values from the 10 minute simulations for the 13 variable problems (see problems 10, 11, and 12, Appendix C) were considered uniformly distributed for each period indicated below.

<u>Time</u>	<u>Period (Problem)</u>	<u>Number of 10 Minute Periods</u>
6:00 A.M.-7:30 A.M.	Off-peak (11)	9
7:30 A.M.-9:30 A.M.	A.M. peak (10)	12
9:30 A.M.-4:00 P.M.	Off-peak (11)	39
4:00 P.M.-6:00 P.M.	P.M. peak (12)	12
6:00 P.M.-8:00 P.M.	Off-peak (11)	12

The savings in seconds per vehicle operating in the six intersection network are determined by:

$$\text{Savings} = \left[(\text{Initial Objective Function Value}) - (\text{Improved Objective Function Value}) \right] \times (\text{Number of 10 Minute Periods})$$

or

$$\text{A.M. Peak Savings} = (33.09 - 28.26) \times (12) = 57.96$$

$$\text{Off-Peak Savings} = (37.43 - 30.48) \times (60) = 417.00$$

$$\text{P.M. Peak Savings} = (32.46 - 24.32) \times (12) = \underline{97.68}$$

$$\text{Total Average Savings for the 10 Minute Periods} = 572.64 \text{ Seconds}$$

With 90,000 vehicles passing through the system during each 14 hour period the number of vehicles per 10 minute period is computed as being:

$$(1/14) \times (1/6) \times (90,000) = 83.48 \text{ vehicles}$$

The total savings in seconds per day for the six intersection sub-network would then be:

$$(572.64) \times (83.48) = 47,803.99 \text{ seconds}$$

Utilizing \$0.00051 as the cost per second of delay (8) the monetary savings for the six intersection subnetwork for the 14 hour day are calculated as being:

$$(47,803.99) \times (\$0.00051) = \$24.38$$

Assuming that the amount of improvement for the six intersection network could be linearly extended to encompass the 77 signalized intersections of the Wichita Falls system, the savings to motorists per day would be:

$$(77/6) \times (\$24.38) = \$312.88$$

Applying the daily savings to a 300 day year, yields an annual savings to motorists of \$93,868.13. This reduction in vehicle operating costs almost doubles the \$51,648.00 figure which was estimated saved by the Wichita Falls system.

Comparison with Alternative Schemes

A comparison was made between the sequential simplex technique

and other existing methodology. Two alternatives were thought suitable for the problem of improving traffic signal settings. The first was a one-factor-at-a-time search scheme. The other was the method of steepest ascent.

One-Factor-at-a-Time Search

Miller and Little (2) applied the one-factor-at-a-time search to the improvement of offsets. They explored two systematic procedures using this technique. One was to vary the absolute offsets of individual signals. The other was to change the relative offset between pairs of signals. However, this method approaches the optimum very slowly, particularly when some mild interaction among the variables is present. Because of this feature, this procedure was rejected for comparative purposes.

Method of Steepest Ascent

The method of steepest ascent used in this thesis falls within the experimental framework of response surface methodology. This method clearly utilizes a form of a sequential approach which directs the progression of experimentation from a poor region to one which in all likelihood contains the optimum (19).

The method of steepest ascent seeks a point on the surface of a hyperplane which, for this research, represents the maximum decrease in the response or objective function as previously discussed (20). This involves the fitting of the following first order model in some region using a 2^k factorial design or a suitable fraction thereof:

$$\hat{y} = b_0 + \sum_{i=1}^k b_i x_i.$$

A minimal \hat{y} , located on the surface of a hypersphere of radius R is sought. This may be denoted by:

$$\text{Minimize } \hat{y} \quad \text{Subject to} \quad \sum_{i=1}^k x_i^2 = R^2$$

Introducing the LaGrange multiplier, λ , this problem becomes:

$$\text{Minimize } Q = b_0 + \sum_{i=1}^k b_i x_i - \lambda \left(\sum_{i=1}^k x_i^2 - R^2 \right)$$

Taking partial derivatives with respect to x_i and λ the following relationships are determined:

$$x_i = \frac{b_i}{2\lambda}, \quad \sum_{i=1}^k x_i^2 = R^2$$

The procedure calls for determining a step size for one of the variables, say x_i . This then fixes the values of λ and R and thus the coordinates of any point along the path of steepest ascent. Responses are observed along this path until curvature in the system is evident. At this point a new first order model is fitted and the process is repeated. When no further reduction in the objective function is noted, the experiment is halted.

For comparative purposes, the six offsets for the A.M. period were the variables selected to be improved. The six variables required

$2^6 = 64$ observations. A one-fourth fraction of a 2^6 design was used. The defining contrasts selected were $x_1 x_2 x_3 x_4 x_5 x_6$ and $x_4 x_5 x_6$ which generated a third contrast $x_1 x_2 x_3$. This combination did not alias any of the main effects with each other.

The variables (offsets) for this problem were arbitrarily assigned a range of four. The variables for this problem were denoted as follows:

<u>Variable</u>	<u>Range</u>
x_1	22 - 26
x_2	72 - 76
x_3	72 - 76
x_4	47 - 51
x_5	47 - 51
x_6	0 - 4
<hr/>	
Coded to:	-1 1

The principle block was used for this experiment. The results of the first trial run are given by the following:

<u>Trial 1</u>	<u>y (objective function)</u>
(1)	35.20
ab	38.60
ac	36.46
bc	36.87
de	33.15

<u>Trial 1</u>	<u>y (objective function)</u>
abde	34.81
acde	32.72
bcde	35.29
df	33.76
abdf	34.15
acdf	34.82
bcd f	33.03
ef	35.52
abef	37.65
acef	35.89
bcef	34.15

The estimates of the parameters in the first order model are:

$$\hat{\underline{\beta}} = (\underline{X}'\underline{X})^{-1} \underline{X}'\underline{y}.$$

Here

$$(\underline{X}'\underline{X})^{-1} = 1/16 \underline{I}_7,$$

$$\underline{X}'\underline{y} = \begin{bmatrix} 562.07 \\ 8.13 \\ 7.03 \\ -3.61 \\ -18.61 \\ -3.71 \\ -24.13 \end{bmatrix}, \text{ and thus } \hat{\underline{\beta}} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} = \begin{bmatrix} 35.129 \\ 0.508 \\ 0.439 \\ -0.225 \\ -1.163 \\ -0.232 \\ -1.508 \end{bmatrix}.$$

A one per cent step in variable x_3 was selected. Coding this step size gave the following:

$$x_3 = \frac{\text{Size of step}}{1/2 \text{ Range}} = \frac{1}{2} = 0.5$$

From the fitted first order model $b_3 = -3.61$. The value of 2λ was then determined as $0.5 = \frac{-3.61}{2\lambda}$ or $2\lambda = -7.22$. From this the following incremental changes were computed for application to the center points:

<u>Variable</u>	<u>Increment</u>
Δx_1	-2.252
Δx_2	-1.974
Δx_3	1.0
Δx_4	5.156
Δx_5	1.028
Δx_6	6.684

The following were the results of incrementing the base coordinates along the path of steepest ascent:

<u>Trial</u>	<u>Response</u>
Base	34.86
Base + Δ	36.66
Base + 2Δ	33.16
Base + 3Δ	35.40

<u>Trial</u>	<u>Response</u>
Base + 4 Δ	31.94
Base + 5 Δ	32.32
Base + 6 Δ	30.16
Base + 7 Δ	35.69
Base + 8 Δ	35.20
Base + 9 Δ to Base + 20 Δ	Greater than the Above

Around the coordinates of the sixth trial (Base + 6 Δ) a new experiment was conducted again using a one-fourth fraction of a 2^6 design. The new variables used were:

<u>Variable</u>	<u>Range</u>	
x_1	8	12
x_2	60	64
x_3	78	82
x_4	78	82
x_5	53	57
x_6	40	44
Coded to:	-1	1

The defining contrasts remained unchanged. The results of the second trial run were:

<u>Trial 2</u>	<u>y (objective function)</u>
(1)	35.41
ab	32.74
ac	34.30
bc	34.64
de	33.62
abde	36.83
acde	36.51
bcde	32.65
df	33.67
abdf	31.72
acdf	38.58
bcd f	36.94
ef	36.11
abef	32.18
acef	35.02
bcef	33.03

The fitting of a second first order model yielded estimates of the following parameters:

$$(X'X)^{-1} = 1/16 \, I_7,$$

$$X'Y = \begin{bmatrix} 553.95 \\ 1.81 \\ -12.49 \\ 9.39 \\ 7.09 \\ -2.05 \\ 0.55 \end{bmatrix}, \text{ and } \hat{\underline{B}} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{bmatrix} = \begin{bmatrix} 34.622 \\ 0.113 \\ -0.781 \\ 0.587 \\ 0.443 \\ 0.128 \\ 0.034 \end{bmatrix}.$$

For a new path of steepest ascent a step length of one per cent was assigned to variables x_3 . The incremental changes assigned were:

<u>Variable</u>	<u>Increment</u>
x_1	0.193
x_2	1.330
x_3	1.0
x_4	0.755
x_5	-0.211
x_6	0.058

The new path of steepest ascent yielded the following:

<u>Trial</u>	<u>Response (Objective Function)</u>
Base	33.81
Base + Δ	32.87
Base + 2Δ	31.58
Base + 3Δ	30.02
Base + 4Δ	31.57

<u>Trial</u>	<u>Response (Objective Function)</u>
Base + 5Δ to Base + 20Δ	Greater than the Above

From the above observations it was obvious that further improvement could not be achieved by the method of steepest ascent. This was caused by the partial derivatives of y evaluated at points close to the optimum being near zero. For this reason the experiment was discontinued as a near optimum region in the factor space had been reached. The coordinates whose $x_3 = 83$ (Base + 3Δ) gave the lowest objective function in value were selected as the improved offsets. With starting offsets = 24 74 74 49 49 2, the improved offsets attained by the method of steepest ascent = 11 66 83 82 54 42. The improvement in the objective function was 4.84 seconds.

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

Simplex Evolutionary Operation was found to be suited for the problem of improving traffic signal settings. A continuation of this research is recommended in the further refining of the sequential simplex procedure and the extension of its use to large networks.

Summary of Results

The empirical adjustment of the simplex side length was a key factor in the successful determination of improved traffic signal settings. This was accomplished by conducting a number of simulation runs each with varying side lengths for the individual problem cases. When criterion for improved signal settings were met, simplex side length adjustment was terminated and the results recorded.

The improvements were shown to be statistically significant. A verification of the lesser improved settings was deemed sufficient for accepting the other settings as being significantly improved.

A cost effectiveness analysis was conducted. A greater efficiency in the overall traffic flow resulting from the improved settings reduced vehicle operating costs appreciably.

A comparison with the method of steepest ascent was made. Although this procedure produced a significant improvement in traffic signal settings, it failed to continue to improve to the extent that the sequential simplex technique did. Further, the simple and automatic

procedure of the sequential simplex, easily adaptable for computer use, could not be matched by the more complex and cumbersome method of steepest ascent.

Conclusions

The conclusions derived from this study are the following:

1. A definite improvement in traffic signal settings had been achieved by the sequential simplex technique. This improvement was verified statistically.
2. A literature survey revealed that, in other techniques employed for the solution of the signal setting problem, the cycle time was either improved independently or was assumed to be optimal. This study clearly showed that the sequential simplex technique was capable of improving the cycle time simultaneously with other settings. This was carried to the extent that improved cycle times were derived for each individual signal in the network.
3. This study revealed that the simplex EVOP procedure of developing improved signal settings is sensitive to side length. This bears further study.
4. An appreciable reduction in vehicle operating costs was achieved when the sequential simplex technique was implemented for the improvement of traffic signal settings.
5. A comparison with the method of steepest ascent favored the sequential simplex technique. The knowledge required by the experimenter in addition to the complex procedures of the method of steepest ascent alone made this improvement scheme unattractive for improving traffic signal settings. The simple, efficient and easily computerized

procedures of the sequential simplex technique found it to be a worthy addition to the present inventory of software available for the improvement of signal settings.

6. One should bear in mind that only improved traffic signal settings were determined. The techniques used in this research were based upon an idealized rather than an exact representation of the real problem. There cannot be any guarantee that the methodology developed will prove to be the best possible solution that could have been implemented for the improvement of traffic signal settings.

Recommendations for Further Study

This research generated a number of areas which require further investigation and clarification. These areas pertain primarily to the application of the sequential simplex to optimizing stochastic systems in general and a continuation of the research presented in this thesis.

Presently there is no provision in the sequential simplex technique for halting the procedure when applied to a stochastic process. When this technique is used to improve a system which is subject to a great amount of error, the experimenter arbitrarily must define criterion under which improved or optimized data are found. It is recommended that further study be conducted in this area for the development of a scheme for terminating the iterative movements of the simplices when applied to a stochastic process.

The sequential simplex was found to be highly sensitive to length of the simplex side length. Many other authors made little mention of this important factor. A great contribution could be made toward

improving the operation of sequential simplex if some systematic method of side length adjustment could be developed. This could perhaps be utilized with a convergence procedure which attempts to terminate the progression of the simplices upon locating optimal or near optimal values.

The experiment conducted in this thesis had shown that the sequential simplex technique was successfully implemented for the improvement of traffic signal settings. Further research in this regard would warrant the application of a modified version of this scheme introduced by Nelder and Mead (23). Their version, the sequential simplex pattern search, could possibly reach the region of optimality much faster than the method used in this thesis. The simplex procedure implemented not only reflects, but also expands and contracts. The set of simplices then adapts itself more effectively to the local landscape elongating down long inclined planes, changing direction upon encountering a valley at an angle and contracting in the neighborhood of a minimum. As a result, when the simplices progress into the region of optimality, convergence takes place and nearly optimal objective function values are achieved.

A direct extension of this thesis would be the application of the methodology used toward a large signalized street network. Since the sequential simplex was found capable of improving settings for a small network, some scheme should be developed for linking an entire signalized traffic network together. One possible approach would be to subdivide a large network into small subnetworks of say six to nine signalized intersections each. After improving the signal settings for

each subnetwork using the methods of this thesis, the linking could take place. A recommended means of accomplishing this would be to first find improved settings among the subnetworks that had common or nearly common values. By considering that the signals with the same setting values were as one, the number of variables would be greatly reduced and a simplex could be constructed for the entire network and tested. In this regard a strategy which could be employed would be to increase the variables by type as was done in the experimental portion of this thesis. Once linking had been accomplished, the sequential simplex technique could then be applied to any traffic control system ranging from a fixed time system to a sophisticated computer controlled system.

APPENDIX A

Computer Listing for the Simplex EVOP Procedure

```

-RUN SIPLX,53S800012,MULLEN-C-J,15
-FOR,IS SIPLX
C SEQUENTIAL SIMPLEX PROCEDURE
C
C LSIZE=NUMBER OF VARIABLES
C X=DESIGN MATRIX
C R(I)=INITIAL ROW VARIABLE COORDINATES
C S(I)=PERCENT GREEN ROW ELEMENTS
C T(I)=OFFSET ROW ELEMENTS
C C(I)=CYCLE ROW ELEMENTS
C ITER=NUMBER OF MATRIX ITERATIONS
C MAX=MAXIMUM ALLOWABLE MATRIX ITERATIONS
C ISTOP=MAXIMUM ALLOWABLE AGE OF COORDINATES
C SIDELN=LENGTH OF A SIDE OF THE SIMPLEX
C KNTR=DIAGONAL COUNTER FOR P VALUES
C IAGE(I)=AGE OF EACH SET OF COORDINATES
C VALUE(I)=RESPONSE CORRESPONDING TO EACH COORDINATE SET
C XNEW=STORAGE OF NEW MATRIX ELEMENTS
C SUM(I)=INTERMEDIATE STEP CONSTANT FOR NEW COORDINATE COMPUTATION
C NEW=INDEX OF COORDINATES SELECTED FOR RECOMPUTATION
C INDEX=ROW IN MATRIX
C NTR=COUNTER FOR NEGATIVE COORDINATES
C KTR=COUNTER FOR MINIMUM GREEN CONSTRAINT
C
  DIMENSION X(20,20),XNEW(20,20),IAGE(20),VALUE(20),NSIM(20),
2SUM(20),R(20)
  DIMENSION GRN(20)
  COMMON/BLK1/ BLKARY(10000)
  COMMON/BLK2/ NTNFLG(2000)
  COMMON/BLK3/ STPFLG(2000)
  COMMON/BLK4/ TRNFLG(2000)
  COMMON/BLK5/ FSTLAN(400)
  COMMON/BLK6/ NOLAN(400)
  COMMON/BLK7/ PLT(400), PSM(400)
  COMMON/BLK8/ FSTBLK(1000)
  COMMON/BLK9/ NOBLKS(1000)

```

```

COMMON/BLOCKA/S(6),DUM1(4),T(6),DUM2(4),C(6),DUM3(4)
5000 FORMAT(I4,F4.0,I4)
6000 FORMAT(1H0,9HNEG COORD)
6001 FORMAT(1H0,17HMAX ITER EXCEEDED)
6002 FORMAT(1H0,I3,13F6.0,F7.2,I4)
6003 FORMAT(1H0,92HROW PGRN1 PGRN2 PGRN3 PGRN4 PGRN5 PGRN6 OFST1 OFST2
20FST3 OFST4 OFST5 OFST6 CYCLE OBJFUN RUN)
6006 FORMAT(1H0,14HNO PLACE TO GO)
6008 FORMAT(1H0,I3,13F6.0)
      READ(5,5000)LSIZE,SIDELN,MAX
      KSIM=-1
      CALL SIMLAT(OBJFUN,KSIM)
      N=LSIZE+1
      G=LSIZE
      ISTOP=N
      DO 5 I=1,6
5      X(1,I)=S(I)*100.
      DO 6 I=7,12
6      X(1,I)=T(I-6)*100.
      X(1,13)=C(1)
      DO 15 J=1,LSIZE
15     R(J)=X(1,J)
      DO 20 I=1,N
20     IAGE(I)=0
      P=SIDELN*(SQRT(G+1.0)-1.0+G)/(G*SQRT(2.0))
      Q=SIDELN*(SQRT(G+1.0)-1.0)/(G*SQRT(2.0))
      ITER=0
      KTR=0
      NTR=0
      NEW=0
      KNTR=1
      DO 30 I=2,N
      DO 31 J=1,LSIZE
31     X(I,J)=Q+R(J)
      X(I,KNTR)=P+R(KNTR)
30     KNTR=KNTR+1

```

```

DO 40 I=1,N
DO 41 J=1,6
S(J)=X(I,J)/100.
T(J)=X(I,J+6)/100.
41 C(J)=X(I,13)
CALL SIMLAT(OBJFUN,KSIM)
NSIM(I)=KSIM
40 VALUE(I)=-OBJFUN
DO 45 I=1,N
DO 45 J=1,LSIZE
45 XNEW(I,J)=X(I,J)
WRITE(6,6003)
50 ITER=ITER+1
DO 51 I=1,N
51 WRITE(6,6002)I,(X(I,J),J=1,LSIZE),VALUE(I),NSIM(I)
IF(ITER.GE.MAX)GO TO 1001
C
C RULE 1 TEST
C
SMALL=VALUE(1)
INDEX=1
DO 60 I=2,N
IF(VALUE(I).GE.SMALL)GO TO 60
SMALL=VALUE(I)
INDEX=I
60 CONTINUE
C
C RULE 3 TEST
C
61 IF(INDEX.EQ.NEW)GO TO 200
C
C RULE 1 PROCEDURE
C
NEW=INDEX
DO 70 I=1,LSIZE
SUM(I)=0.0

```



```

DO 70 J=1,N
70 SUM(I)=SUM(I)+X(J,I)
DO 80 I=1,LSIZE
80 XNEW(INDEX,I)=(2.0/G)*SUM(I)-((G+2.0)/G)*X(INDEX,I)
DO 85 J=1,6
GRN(J)=(1.-XNEW(INDEX,J)/100.)*XNEW(INDEX,13)
C
C MINIMUM GREEN TEST
C
IF(GRN(J).LT.18.)GO TO 300
85 CONTINUE
C
C NEGATIVE COORDINATE TEST
C
DO 87 K=1,LSIZE
IF(XNEW(INDEX,K).GE.0.0)GO TO 87
C
C NEGATIVE COORDINATE PROCEDURE
C
NTR=NTR+1
IF(NTR.GE.N)GO TO 6005
WRITE(6,6000)
WRITE(6,6008)INDEX,(XNEW(INDEX,L),L=1,LSIZE)
DO 95 L=1,LSIZE
95 XNEW(INDEX,L)=X(INDEX,L)
GO TO 200
87 CONTINUE
DO 90 I=1,N
IAGE(I)=IAGE(I)+1
IF(I.EQ.INDEX)IAGE(I)=0
C
C RULE 2 TEST
C
IF(IAGE(I).LT.ISTOP)GO TO 90
C
C RULE 2 PROCEDURE

```

C

```

      DO 81 L=1,6
      S(L)=X(I,L)/100.
      T(L)=X(I,L+6)/100.
81  C(L)=X(I,13)
      CALL SIMLAT(OBJFUN,KSIM)
      NSIM(I)=KSIM
      VALUE(I)=-OBJFUN
      IAGE(I)=0
90  CONTINUE
      DO 100 I=1,N
      DO 100 J=1,LSIZE
100  X(I,J)=XNEW(I,J)
      KTR=0
      NTR=0
      DO 110 I=1,6
      S(I)=X(INDEX,I)/100.
      T(I)=X(INDEX,I+6)/100.
110  C(I)=X(INDEX,13)
      CALL SIMLAT(OBJFUN,KSIM)
      NSIM(INDEX)=KSIM
      VALUE(INDEX)=-OBJFUN
      GO TO 50
300  KTR=KTR+1
      IF(KTR.GE.N) GO TO 6005
      WRITE(6,700_)
7000 FORMAT(1H0,18HMIN GREEN VIOLATED)
      WRITE(6,6008)INDEX,(XNEW(INDEX,L),L=1,LSIZE)
      DO 310 L=1,LSIZE
310  XNEW(INDEX,L)=X(INDEX,L)
C
C  RULE 3 PROCEDURE
C
200  GYP=VALUE(INDEX)
      VALUE(INDEX)=1.0E+36
      SMALL=VALUE(1)

```

```
      IN=1
      DO 210 I=2,N
      IF(VALUE(I).GE.SMALL)GO TO 210
      SMALL=VALUE(I)
      IN=I
210  CONTINUE
      VALUE(INDEX)=GYP
      INDEX=IN
      GO TO 61
6005 WRITE(6,6006)
      STOP
1001 WRITE(6,6001)
      STOP
      END
```

APPENDIX B

Computer Listing for the Traffic Simulation and Sample Data

```

-FOR, IS SIMLAT
  SUBROUTINE SIMLAT(OBJFUN,KRUNS)
  INTEGER BLKARY, STPFLG, TRNFLG, FSTLAN, FSTBLK, RLINC, SLINC
  INTEGER SGAP , RGAP, LGAP
  REAL MAXTIM
  COMMON/BLOCKA/PGRN(10),OFFSET(10),CYCLE(10)
  COMMON/BLK1/ BLKARY(10000)
  COMMON/BLK2/ NTNFLG(2000)
  COMMON/BLK3/ STPFLG(2000)
  COMMON/BLK4/ TRNFLG(2000)
  COMMON/BLK5/ FSTLAN(400)
  COMMON/BLK6/ NOLAN(400)
  COMMON/BLK7/ PLT(400), PSM(400)
  COMMON/BLK8/ FSTBLK(1000)
  COMMON/BLK9/ NOBLKS(1000)
  DIMENSION LLINC(400), LNCSEG(400,4), RLINC(400), SLINC(400),
1  IDNO(400,2), LNCOP(400), LNCRT(400), LNCLT(400),
2  LCNO(1000), NOSTPS(1000), NOWATS(1000), LNKLAN(1000),
3  INTTYP(1000), NOTYP(1000), CLOKGN(25), CLOKLT(100),
4  LSTATE(100), KARY(25), IDNO(150), NXNODE(50),
5  LINTER(50), PARAMS(50,4), LNCGEN(50),
6  NOINT(150), LNCSP(50,4), NCARWT(50)
100 FORMAT(1H )
101 FORMAT(I4,I2,I4, F6.0)
102 FORMAT(10I4)
103 FORMAT(I4,1X,I4,I2,7I4,1X,I4,1X,I4,I2,7I4)
104 FORMAT(1H1,-INPUT DATA CARDS-,//,7X,-GENERATE AND TERMINATE NODES-
2,///,21X,-NODE NUMBER-,5X,-TYPE-,6X,-AVG. TIME BETWEEN ARRIVALS-,/
3/)
105 FORMAT(26X,I3,10X,I2,14X, F10.3)
106 FORMAT(1H1,6X,-INTERSECTION NODES-,///,20X,-NODE NUMBER SIGNALIZ
1ED PRIMARY NODES SECONDARY NODES IF SIGNALIZED -,/,82X,-CYCLE
2 PERCENT RED - GREEN OFFSET-,//)
107 FORMAT(23X,I4,11X,-YES-,7X,I4,- --,I4,6X,I4,- --,I4,8X,I4,11X,
2 F4.3,- - -,F4.3,3X,F4.3)
108 FORMAT(23X,I4,11X,-NO -,7X,I4,- --,I4,6X,I4,- --,I4,8X,I4)

```

```

109 FORMAT(1H1,--STREET SEGMENTS--,///,20X,--NODE NUMBERS LANES LENGTH
1 PROBABILITIES AND NEXT NODES--,/,42X,--(FT) LEFT-,16X,--STRAIGHT
2-,12X,--RIGHT-)
110 FORMAT(20X,I4,-- --,I4,6X,I1,5X,I4,4X,3(F4.3,I7,9X))
111 FORMAT(3F10.0)
130 FORMAT(1H1)
131 FORMAT(20X,2I10,F10.5)
132 FORMAT(I4,I10,2I5,2I10,2I5,I10,2I5,F10.3,F5.3)
133 FORMAT(I4,I10,I5,4I10)
134 FORMAT(I4,I10,I10,3I5,F10.1,2F10.3)
150 FORMAT(20(1X,1H*),49H THE RED AND GREEN PHASE DO NOT EQUAL 100 PER
1CENT ,)
151 FORMAT (1H1,//////////,40(3H **),/,40X,-- THE PROGRAM EXCEEDED THE
2NUMBER OF VEHICLES PERMITTED.--,///,40(3H **),/)
152 FORMAT (20(1X,1H*), 38H THE PROGRAM STOPPED AT LABEL NUMBER ,
2 I6)
153 FORMAT(2H2,//////////,25(2H**,3X),/,- THIS RUN MAY NOT GIVE VALI
2D RESULTS DUE TO THE CALCULATED BLOCK LENGTH (IN FEET) -, I5)
KRUNS = KRUNS + 1
IF (KRUNS.NE.0) GO TO 1000

```

C THIS SECTION IS ONLY ENTERED ONCE TO INITIALIZE NETWORK.

```

KRN = 1 + KRUNS
NBLKS = 1
KCARS = 2000
DO 5 I=1,KCARS
5 STPFLG(I) = -999
READ(5,111) AVSPD,TIMINT,MAXTIM
MAXTIM = MAXTIM * 60.0
LGAP = 8.0 / TIMINT
RGAP = 8.0 / TIMINT
SGAP = 8.0 / TIMINT
LENGTH = AVSPD * TIMINT * 5280.0 / 3600.0
IF (LENGTH.LT.16 .OR.LENGTH.GT.30 ) WRITE (6,153) LENGTH
WRITE(6,104)

```

10 K = K + 1

C C

C READ DATA CARDS ON ALL GENERATE AND TERMINATE NODES.
C IF KTYPE = -1 -- TERMINATE ONLY NODE
C = +1 -- GENERATE ONLY NODE
C = 0 -- BOTH

C C

READ(5,101) IDNO(K), KTYPE, NXNODE(K), DELT
IF (IDNO(K).GE.9999) GO TO 31
IF (KTYPE.EQ.+1) GO TO 20

C TERMINATE FILE ESTABLISHMENT

KTERS = KTERS + 1
LNTER(KTERS) = K
IF (KTYPE.NE.-1) GO TO 20
WRITE(6,105) IDNO(K),KTYPE
GO TO 10
20 CONTINUE
WRITE(6,105) IDNO(K),KTYPE,DELT

C GENERATE FILE ESTABLISHMENT

KGEN = KGEN + 1
LNCGEN(KGEN) = K
PARAMS(KGEN,1) = DELT
GO TO 10
31 CONTINUE
WRITE(6,106)
K = K - 1

32 K = K + 1

C C C C C C C C C C C C C C C C C C

C READ DATA CARDS ON ALL INTERSECTIONS.
C KTYPE = 1 IF SIGNALIZED INTERSECTIONS
C = 0 OTHERWISE

C C C C C C C C C C C C C C C C C C

READ (5,102) IDNO(K), KTYPE, (KARY(J),J=1,8)
IF (IDNO(K).EQ.9999) GO TO 40
KTHINT = KTHINT + 1
IF (KTYPE.EQ.1) GO TO 35

C NON SIGNALIZED INTERSECTIONS
NUSTP = NUSTP + 1
WRITE(6,108) IDNO(K), (KARY(J),J=1,4)
NOINT(KTHINT) = -K
DO 33 I=1,4
33 LNCSP(NUSTP,I) = -KARY(I)
GO TO 32

C SIGNALIZED INTERSECTIONS

35 NULIT = NULIT + 1
A = FLOAT(KARY(6)) / 1000.
B = FLOAT(KARY(7)) / 1000.
C = FLOAT(KARY(8)) / 1000.
WRITE(6,107) IDNO(K), (KARY(J),J=1,5),A,B,C
NOINT(KTHINT) = K
LSTATE(NULIT) = -1
CYCLE(NULIT) = KARY(5)
PGRN(NULIT) = FLOAT(KARY(7)) / 1000.


```

      K1 = KARY(6) + KARY(7)
      IF (K1.NE.1000) WRITE(6,150)
      CLOKLT(NULIT) = CYCLE(NULIT)*FLOAT(KARY(8)) / 1000.
      OFFSET(NULIT) = C
      DO 36 I=1,4
36  LNCSEG(NULIT,I) = -KARY(I)
      GO TO 32

```

C READ DATA CARDS ON ALL STREET SEGMENTS

```

40  CONTINUE
      KNTINS = KTHINT
      KLINKS = 0
      WRITE(6,109)
41  CONTINUE
      IF (KLINKS.GE.45) GO TO 40
      READ (5,103) (KARY(J),J=1,20)
      IF (KARY(1).EQ.9999) GO TO 60
      WRITE(6,100)
      A = FLOAT(KARY(5)) / 1000.
      B = FLOAT(KARY(7)) / 1000.
      C = FLOAT(KARY(9)) / 1000.
      WRITE(6,110) (KARY(J),J=1,4),A,KARY(6),B,KARY(8),C,KARY(10)
      KLINKS = KLINKS + 2
      KLINKS = KLINKS + 1
      IDNO(KLINKS,1) = KARY(1)
      IDNO(KLINKS,2) = KARY(2)
      NOLAN(KLINKS) = KARY(3)
      FSTLAN(KLINKS) = KLINKS + 1
      NUMBKS = KARY(4)/LENGTH + 1
      PLT(KLINKS) = A
      PSM(KLINKS) = A + B
      LLINC(KLINKS) = -KARY(6)
      SLINC(KLINKS) = -KARY(8)
      RLINC(KLINKS) = -KARY(10)
      KNT = KARY(3)

```

```

      IF (KNT-3) 43,44,42
42 KNT = 7
43 KNT = KNT - 1
44 KNT = KNT - 1
      K1 = NOLAN(KLINKS)
      DO 45 I=1,K1
      KNT = KNT + 1
      KLANES = KLANES + 1
      FSTBLK(KLANES) = NBLKS + 1
      NOBLKS(KLANES) = NUMBKS
      NBLKS = NBLKS + NUMBKS
      LCNO(KLANES) = KLINKS
45 LNKLAN(KLANES) = KNT

```

C IF STREET SEGMENT IS ONE WAY GO BACK AND READ ANOTHER CARD

```

      IF (KARY(11).EQ.0) GO TO 41
      A = FLOAT(KARY(15)) / 1000.
      B = FLOAT(KARY(17)) / 1000.
      C = FLOAT(KARY(19)) / 1000.
      WRITE(6,110) (KARY(J),J=11,14),A,KARY(16),B,KARY(18),C,KARY(20)
      KLINKS = KLINKS + 1
      KLINKS = KLINKS + 1
      IDNO(KLINKS,1) = KARY(11)
      IDNO(KLINKS,2) = KARY(12)
      NOLAN(KLINKS) = KARY(13)
      FSTLAN(KLINKS) = KLANES + 1
      NUMBKS = KARY(14) / LENGTH + 1
      PLT(KLINKS) = A
      PSM(KLINKS) = A + B
      LLINC(KLINKS) = -KARY(16)
      SLINC(KLINKS) = -KARY(18)
      RLINC(KLINKS) = -KARY(20)
      KNT = KARY(13)
      IF (KNT-3) 53,54,52
52 KNT = 7

```

```

53 KNT = KNT - 1
54 KNT = KNT - 1
   K1 = NOLAN(KLINKS)
   DO 55 I=1,K1
   KNT = KNT + 1
   KLANES = KLANES + 1
   FSTBLK(KLANES) = NBLKS + 1
   NOBLKS(KLANES) = NUMBKS
   NBLKS = NBLKS + NUMBKS
   LCNO(KLANES) = KLINKS
55 LNKLAN(KLANES) = KNT
   GO TO 41
60 DO 81 KTHLNK=1,KLINKS
   ITH = LLINC(KTHLNK)
   JTH = SLINC(KTHLNK)
   KTH = RLINC(KTHLNK)
   IF (ITH.EQ.0.AND.JTH.EQ.0.AND.KTH.EQ.0) GO TO 81
   KAUX = 0
   KA = ITH*JTH*KTH
   IF (KA.EQ.0) KAUX = 2
   DO 80 KTST=1,KLINKS
   IF(KTST - KTHLNK) 61,80,61
61 IF (IDNO(KTHLNK,2)-IDNO(KTST,1)) 70,62,70
62 KCHK = IDNO(KTST,2)
   IF (ITH + KCHK) 64,63,64
63 KAUX = KAUX + 1
   LLINC(KTHLNK) = KTST
   GO TO 79
64 IF (JTH + KCHK) 66,65,66
65 KAUX = KAUX + 1
   SLINC(KTHLNK) = KTST
   GO TO 79
66 IF (KTH + KCHK) 80,67,80
67 KAUX = KAUX + 1
   RLINC(KTHLNK) = KTST

```

```

      GO TO 79
70 IF (IDNO(KTHLNK,2)-IDNO(KTST,2)) 80,71,80
71 KCHK = IDNO(KTST,1)
   IF (ITH + KCHK) 74,73,74
73 KAUX = KAUX + 1
   LNCLT(KTHLNK) = KTST
   GO TO 79
74 IF (JTH + KCHK) 76,75,76
75 KAUX = KAUX + 1
   LNCOP(KTHLNK) = KTST
   GO TO 79
76 IF (KTH + KCHK) 80,77, 80
77 KAUX = KAUX + 1
   LNCRT(KTHLNK) = KTST
79 IF (KAUX.EQ.6) GO TO 81
80 CONTINUE
   IF (LLINC(KTHLNK).LE.0) LLINC(KTHLNK) = 0
   IF (SLINC(KTHLNK).LE.0) SLINC(KTHLNK) = 0
   IF (RLINC(KTHLNK).LE.0) RLINC(KTHLNK) = 0
81 CONTINUE
   DO 210 KT=1,KTERS
   KAUX = LENTER(KT)
   ITH = IDNO(KAUX)
   JTH = NXNODE(KAUX)
   DO 205 LNK=1,KLINKS
   IF (ITH.EQ.IDNO(LNK,2).AND.JTH.EQ.IDNO(LNK,1)) GO TO 206
205 CONTINUE
   STOP
206 LENTER(KT) = LNK
210 CONTINUE
   DO 220 KG=1,KGEN
   KAUX = LNCGEN(KG)
   ITH = IDNO(KAUX)
   JTH = NXNODE(KAUX)
   DO 215 LNK=1,KLINKS
   IF (ITH.EQ.IDNO(LNK,1).AND.JTH.EQ.IDNO(LNK,2)) GO TO 216

```

```

215 CONTINUE
    STOP
216 LNCGEN(KG) = LNK
220 CONTINUE
    DO 260 INT=1,KNTINS
        IF (NOINT(INT).LE.0) GO TO 235

```

C SIGNALIZED INTERSECTIONS

```

    KSIGS = KSIGS + 1
    K = NOINT(INT)
    NINT = IDNO(K)
    K = 0
    K1 = LNCSG(KSIGS,1)
    K2 = LNCSG(KSIGS,2)
    K3 = LNCSG(KSIGS,3)
    K4 = LNCSG(KSIGS,4)
    K5 = K1 * K2 * K3 * K4
    IF (K5.EQ.0) K=1
    DO 230 LNK=1,KLINKS
        IF (IDNO(LNK,2).NE.NINT) GO TO 230
        IF (IDNO(LNK,1) + K1) 222,221,222
221 KAUX = KSIGS
        LNCSG(KSIGS,1) = LNK
        GO TO 228
222 IF (IDNO(LNK,1) + K2) 224,223,224
223 KAUX = KSIGS
        LNCSG(KSIGS,2) = LNK
        GO TO 228
224 IF (IDNO(LNK,1) + K3) 226,225,226
225 KAUX = -KSIGS
        LNCSG(KSIGS,3) = LNK
        GO TO 228
226 IF (IDNO(LNK,1) + K4) 230,227,230
227 KAUX = -KSIGS
        LNCSG(KSIGS,4) = LNK

```

```

228 KFST = FSTLAN(LNK)
   KLST = FSTLAN(LNK) + NOLAN(LNK) - 1
   DO 229 LAN=KFST,KLST
     INTTYP(LAN) = +1
229 NOTYP(LAN) = KAUX
   K = K + 1
   IF (K.EQ.4) GO TO 260
230 CONTINUE
   DO 231 K=1,4
     IF (LNCSG(KSIGS,K).LT.0) LNCSG(KSIGS,K) = 0
231 CONTINUE
   GO TO 260
235 CONTINUE

```

C NONSIGNALIZED INTERSECTIONS

```

KSTPS = KSTPS + 1
K = -NOINT(INT)
NINT = IDNO(K)
K = 0
K1 = LNCSP(KSTPS,1)
K2 = LNCSP(KSTPS,2)
K3 = LNCSP(KSTPS,3)
K4 = LNCSP(KSTPS,4)
K5 = K1 * K2 * K3 * K4
IF (K5.EQ.0) K=1
DO 250 LNK=1,KLINKS
  IF (IDNO(LNK,2).NE.NINT) GO TO 250
  IF (IDNO(LNK,1) + K1) 242,241,242
241 KAUX = KSTPS
   LNCSP(KSTPS,1) = LNK
   GO TO 248
242 IF (IDNO(LNK,1) + K2) 244,243,244
243 KAUX = KSTPS
   LNCSP(KSTPS,2) = LNK
   GO TO 248

```

```

244 IF (IDNO(LNK,1) + K3) 246,247,248
245 KAUX = -KSTPS
    LNCSP(KSTPS,3) = LNK
    GO TO 248
246 IF (IDNO(LNK,1) + K4) 250,247,250
247 KAUX = -KSTPS
    LNCSP(KSTPS,4) = LNK
248 KFST = FSTLAN(LNK)
    KLST = FSTLAN(LNK) + NOLAN(LNK) - 1
    DO 249 LAN=KFST,KLST
        INTTYP(LAN) = -1
249 NOTYP(LAN) = KAUX
    K = K + 1
    IF (K.EQ.4) GO TO 260
250 CONTINUE
    DO 251 K=1,4
        IF (LNCSP(KSTPS,K).LT.0) LNCSP(KSTPS,K) = 0
251 CONTINUE
260 CONTINUE
    DO 280 K=1,KTERS
        LNK = LINTER(K)
280 PSM(LNK) = 1.0
        NTRMS = KTERS
        NGENS = KGEN
        WRITE(6,130)
        GO TO 2000
1000 CLOK = 0
    DO 1020 K=1,NGENS
        CLOKGN(K) = 0
1020 CONTINUE
    DO 1040 K=1,6
        CLOKLT(K) = OFFSET(K) * CYCLE(K)
        LSTATE(K) = -1
1040 CONTINUE

```

C C C C C C C C C C C C

```

C      THIS SECTION IS ENTERED EACH TIME TO CHANGE THE SIGNAL
C      LIGHTINGS OF THOSE INTERSECTIONS WHICH ARE DUE TO BE
C      CHANGED.

```

```

C      C      C      C      C      C      C      C      C      C      C      C      C      C

```

```

2000 CLOK = CLOK + TIMINT
      IF (CLOK.GE.MAXTIM) GO TO 7000
      DO 2200 KTHLT=1,KSIGS
      IF (CLOK.LT.CLOKLT(KTHLT)) GO TO 2200
      AUX = LSTATE(KTHLT)
      CYC = CYCLE(KTHLT)
      CLOKLT(KTHLT) = CLOKLT(KTHLT) + CYC * (0.5 + AUX*(0.5-PGRN(KTHLT)))
      LSTATE(KTHLT) = (-1)*LSTATE(KTHLT)
      IF (LSTATE(KTHLT).EQ.1) GO TO 2100

```

```

C      FOR THIS PATH THE LIGHT HAS JUST TURNED RED FOR PRIMARY.

```

```

      KTHLK = LNCSG(KTHLT,1)
      IF (KTHLK.EQ.0) GO TO 2050
      LANE = FSTLAN(KTHLK) + NOLAN(KTHLK) - 1
      NBLK = FSTBLK(LANE)
      IF (BLKARY(NBLK).EQ.0) GO TO 2050
      NVEH = BLKARY(NBLK)
      IF (TRNFLG(NVEH).NE.1) GO TO 2050
      NXTLK = LLINC(KTHLK)
      CALL SNEAK(NXTLK,NBLK,NVEH,KRN)

```

```

2050 KTHLK = LNCSG(KTHLT,2)
      IF (KTHLK.EQ.0) GO TO 2200
      LANE = FSTLAN(KTHLK) + NOLAN(KTHLK) - 1
      NBLK = FSTBLK(LANE)
      IF (BLKARY(NBLK).EQ.0) GO TO 2200
      NVEH = BLKARY(NBLK)
      IF (TRNFLG(NVEH).NE.1) GO TO 2200
      NXTLK = LLINC(KTHLK)

```



```
CALL SNEAK(NXTLK,NBLK,NVEH,KRN)
GO TO 2200
```

```

2100 KTHLK = LNCSG(KTHLT,3)
      IF (KTHLK.EQ.0) GO TO 2150
      LANE = FSTLAN(KTHLK) + NOLAN(KTHLK) - 1
      NBLK = FSTBLK(LANE)
      IF (BLKARY(NBLK).EQ.0) GO TO 2150
      NVEH = BLKARY(NBLK)
      IF (TRNFLG(NVEH).NE.1) GO TO 2150
      NXTLK = LLINC(KTHLK)
      CALL SNEAK(NXTLK,NBLK,NVEH,KRN)
2150 KTHLK = LNCSG(KTHLT,4)
      IF (KTHLK.EQ.0) GO TO 2200
      LANE = FSTLAN(KTHLK) + NOLAN(KTHLK) - 1
      NBLK = FSTBLK(LANE)
      IF (BLKARY(NBLK).EQ.0) GO TO 2200
      NVEH = BLKARY(NBLK)
      IF (TRNFLG(NVEH).NE.1) GO TO 2200
      NXTLK = LLINC(KTHLK)
      CALL SNEAK(NXTLK,NBLK,NVEH,KRN)
2200 CONTINUE

```

C C

C THIS SECTION GENERATES TRAFFIC AND THEN PLACES IT IN THE
C NETWORK.

C C

```

DO 3500 KGEN=1,NGENS
3050 IF (CLOK.LT.CLOKGN(KGEN)) GO TO 3250
NCARWT(KGEN) = NCARWT(KGEN) + 1
RN = RN1(KRN)

```

```

      TIM = (ALOG(RN) * PARAMS(KGEN,1)) * (-1)
3200 CLOKGN(KGEN) = CLOKGN(KGEN) + TIM
      GO TO 3050
3250 IF (NCARWT(KGEN).EQ.0) GO TO 3500
      DO 3300 KAUX = 1,KCARS
      IF (STPFLG(KAUX).EQ.-999) GO TO 3320
3300 CONTINUE
      WRITE (6,151)
      STOP
3320 NVEH = KAUX
      NXTLK = LNCGEN(KGEN)
      NBLK = 1
      CALL SNEAK(NXTLK,NBLK,NVEH,KRN)
      IF (STPFLG(NVEH).EQ.-999) GO TO 3500
      NOVEH=NOVEH+1
      NOVEIN=NOVEIN+1
      NCARWT(KGEN) = NCARWT(KGEN) - 1
      STPFLG(NVEH) = 0
      GO TO 3250
3500 CONTINUE

```

```

C      C      C      C      C      C      C      C      C      C      C      C      C      C

```

```

C      THIS SECTION EXAMINES ALL LANES TO SEE IF THERE IS A
C      VEHICAL THAT IS READY TO EXIT THE LINK AND ENTER A NEW
C      LINK.  THE MOVEMENT IS THEN MADE.

```

```

C      C      C      C      C      C      C      C      C      C      C      C      C      C

```

```

      DO 4500 KTHLAN=1,KLANES
      IF (INTTYP(KTHLAN).EQ.0) GO TO 4500
      NBLK = FSTBLK(KTHLAN)
      IF (BLKARY(NBLK).EQ.0) GO TO 4500
      NVEH = BLKARY(NBLK)

```

IF (INTTYP(KTHLAN).NE.1) GO TO 4250

C THIS SECTION IS FOR SIGNALIZED INTERSECTIONS.

NSIG = NOTYP(KTHLAN)

IF (NSIG.GT.0) GO TO 4019

NSIG = -NSIG

IF (LSTATE(NSIG).NE.-1) GO TO 4490

GO TO 4020

4019 IF (LSTATE(NSIG).NE.1) GO TO 4490

4020 CONTINUE

KTHLK = LCNO(KTHLAN)

IF (TRNFLG(NVEH)) 4025,4026,4027

4025 NXTLK = RLINC(KTHLK)

GO TO 4028

4026 NXTLK = SLINC(KTHLK)

GO TO 4028

4027 NXTLK = LLINC(KTHLK)

4028 CONTINUE

IF (NTNFLG(NVEH).NE.999) GO TO 4030

RN = RN1(KRN)

IF (RN.LE.PLT(NXTLK)) NTNFLG(NVEH) = 1

IF (RN.GT.PLT(NXTLK).AND.RN.LE.PSM(NXTLK)) NTNFLG(NVEH) = 0

IF (RN.GT.PSM(NXTLK)) NTNFLG(NVEH) = -1

4030 CONTINUE

IF (TRNFLG(NVEH).NE.+1) GO TO 4050

IF (LNCOP(KTHLK).EQ.0) GO TO 4050

LNCOP1 = LNCOP(KTHLK)

K1 = NOLAN(LNCOP1)

DO 4040 KCHK=1,K1

LANCHK = FSTLAN(LNCOP1) + KCHK - 1

K2 = FSTBLK(LANCHK)

K3 = K2 + SGAP - 1

DO 4040 KCHBLK=K2,K3

IF (BLKARY(KCHBLK).NE.0) GO TO 4490

4040 CONTINUE

```

4050 CONTINUE
      CALL CHKMV(NBLK,NXTLK,NTNFLG(NVEH),KCHECK,NVEH)
      IF (KCHECK) 4500,4500,4490
4250 CONTINUE

```

C THIS SECTION IS FOR NON SIGNALIZED INTERSECTIONS

```

      NOINT = NOTYP(KTHLAN)
      IF (NOINT.LT.0) GO TO 4350

```

C THIS SECTION IS FOR NON SIGNALIZED, MAIN INTERSECTIONS

```

      KTHLK = LCNO(KTHLAN)
      IF (TRNFLG(NVEH)) 4255,4256,4257
4255 NXTLK=RLINC(KTHLK)
      GO TO 4258
4256 NXTLK=SLINC(KTHLK)
      GO TO 4258
4257 NXTLK=LLINC(KTHLK)
4258 CONTINUE
      IF (NTNFLG(NVEH).NE.999) GO TO 4260
      RN = RN1(KRN)
      IF (RN.LE.PLT(NXTLK)) NTNFLG(NVEH) = 1
      IF (RN.GT.PLT(NXTLK).AND.RN.LE.PSM(NXTLK)) NTNFLG(NVEH) =
      IF (RN.GT.PSM(NXTLK)) NTNFLG(NVEH) = -1
4260 CONTINUE
      IF (TRNFLG(NVEH).NE.+1) GO TO 4280
      LNCOP1 = LNCOP(KTHLK)
      IF (LNCOP1.EQ.0) GO TO 4280
      K1 = NOLAN(LNCOP1)
      DO 4270 KCHK=1,K1
      LANCHK = FSTLAN(LNCOP1) + KCHK - 1
      K2 = FSTBLK(LANCHK)
      K3 = K2 + LGAP - 1
      DO 4270 KCHBLK=K2,K3
      IF (BLKARY(KCHBLK).NE.0) GO TO 4490
4270 CONTINUE

```

```

4280 CONTINUE
      CALL CHKMV(NBLK,NXTLK,NTNFLG(NVEH),KHECK,NVEH)
      IF (KHECK) 4500,4500,4490
4350 CONTINUE

C      THIS SECTION IS FOR A NON SIGNALIZED, SECONDARY, INTERSECTION

      IF (STPFLG(NVEH)) 4355,4490,4355
4355 NOINT = - NOINT
      NVEH = BLKARY(NBLK)
      KTHLK = LCNO(KTHLAN)
      IF (TRNFLG(NVEH)) 4360,4400,4410

C      THIS PATH IS FOR NON-SIGNALIZED, SECONDARY, RIGHT TURNING TRAFFIC.

4360 NXTLK = RLINC(KTHLK)
      IF (NTNFLG(NVEH).NE.999) GO TO 4370
      RN = RN1(KRN)
      IF (RN.LE.PLT(NXTLK)) NTNFLG(NVEH) = 1
      IF (RN.GT.PLT(NXTLK).AND.RN.LE.PSM(NXTLK)) NTNFLG(NVEH) = 0
      IF (RN.GT.PSM(NXTLK)) NTNFLG(NVEH) = -1
4370 CONTINUE
      LNCLTI = LNCLT(KTHLK)
      IF (LNCLTI.EQ.0) GO TO 4390
      K1 = NOLAN(LNCLTI)
      DO 4380 KCHK=1,K1
      LANCHK = FSTLAN(LNCLTI) + KCHK - 1
      K2 = FSTBLK(LANCHK)
      K3 = K2 + RGAP - 1
      DO 4380 KCHBLK=K2,K3
      IF (BLKARY(KCHBLK).NE.0) GO TO 4490
4380 CONTINUE
4390 CONTINUE
      CALL CHKMV(NBLK,NXTLK,NTNFLG(NVEH),KHECK,NVEH)
      IF (KHECK) 4500,4500,4490

```

C THIS PATH FOR NON-SIGNALIZED, STRAIGHT, SECONDARY TRAFFIC

4400 NXTLK = SLINC(KTHLK)
GO TO 4411

C THIS PATH FOR NON-SIGNALIZED, LEFT TURNING, SECONDARY TRAFFIC

4410 NXTLK = LLINC(KTHLK)
4411 CONTINUE
IF (NTNFLG(NVEH).NE.999) GO TO 4420
RN = RN1(KRN)
IF (RN.LE.PLT(NXTLK)) NTNFLG(NVEH) = 1
IF (RN.GT.PLT(NXTLK).AND.RN.LE.PSM(NXTLK)) NTNFLG(NVEH) = 0
IF (RN.GT.PSM(NXTLK)) NTNFLG(NVEH) = -1

C THE LEFT TURNING TRAFFIC AND THE STRAIGHT TRAFFIC BOTH MUST CHECK
C GAPS FROM BOTH DIRECTIONS ON SIDE STREETS. THE LEFT MUST ALSO
C CHECK ONCOMING TRAFFIC.

4420 CONTINUE
LNCLTI = LNCLT(KTHLK)
IF (LNCLTI.EQ.0) GO TO 4440
K1 = NOLAN(LNCLTI)
DO 4430 KCHK=1,K1
LANCHK = FSTLAN(LNCLTI) + KCHK - 1
K2 = FSTBLK(LANCHK)
K3 = K2 + LGAP - 1
DO 4430 KCHBLK=K2,K3
IF (BLKARY(KCHBLK).NE.0) GO TO 4490

4430 CONTINUE

4440 CONTINUE
LNCRTI = LNCRT(KTHLK)
IF (LNCRTI.EQ.0) GO TO 4460
K5 = NOLAN(LNCRTI)
DO 4450 KCHK=1,K5
LANCHK = FSTLAN(LNCRTI) + KCHK - 1

```

      K6 = FSTBLK(LANCHK)
      K7 = K6 + RGAP - 1
      DO 4450 KCHBLK=K6,K7
        IF (BLKARY(KCHBLK).NE.0) GO TO 4490
4450 CONTINUE
4460 CONTINUE
      CALL CHKMV(NBLK,NXTLK,NTNFLG(NVEH),KHECK,NVEH)
      IF (KHECK.NE.1) GO TO 4500
4490 CONTINUE
      IF (STPFLG(NVEH).NE.0) GO TO 4495
      NOSTPS(KTHLAN) = NOSTPS(KTHLAN) + 1
      STPFLG(NVEH) = 1
4495 CONTINUE
      NOWATS(KTHLAN) = NOWATS(KTHLAN) + 1
4500 CONTINUE

```

```

C      C      C      C      C      C      C      C      C      C      C      C      C      C
C      THIS SECTION MOVES TRAFFIC WITHIN A LANE
C      C      C      C      C      C      C      C      C      C      C      C      C      C

```

```

      DO 5500 KTHLA=1,KLANES
        IF (NOBLKS(KTHLA) - 2) 5500,5401,5010
5010 KB1 = FSTBLK(KTHLA) + 1
        KBNTL = FSTBLK(KTHLA) + NOBLKS(KTHLA) - 2
        LANE = LNKLAN(KTHLA)
        DO 5400 NBLK=KB1,KBNTL
          IF (BLKARY(NBLK).EQ.0) GO TO 5400
          NVEH = BLKARY(NBLK)
          IF (BLKARY(NBLK-1).NE.0) GO TO 5100
          IF (STPFLG(NVEH)) 5050,5050,5045
5045 STPFLG(NVEH) = -1
          GO TO 5395

```

```

5050 CONTINUE
    BLKARY(NBLK) = 0
    KAUX = NBLK - 1
    BLKARY(KAUX) = NVEH
    STPFLG(NVEH) = 0
    GO TO 5400
5100 CONTINUE
    IF (TRNFLG(NVEH).NE.0) GO TO 5390
    IF (LANE.EQ.0) GO TO 5390
    IF (LANE.GT.2) GO TO 5120
    IF (LANE.EQ.1) CALL CHLT(NVEH,NBLK,KTHLA,KHECK)
    IF (LANE.EQ.2) CALL CHRT(NVEH,NBLK,KTHLA,KHECK)
    IF (KHECK) 5400,5400,5390
5120 CONTINUE
    IF (LANE.EQ.3.OR.LANE.EQ.6) GO TO 5130
    IF (LANE.EQ.5.OR.LANE.EQ.9) GO TO 5140
    CALL CHRT(NVEH,NBLK,KTHLA,KHECK)
    IF (KHECK.NE.1) GO TO 5400
    CALL CHLT(NVEH,NBLK,KTHLA,KHECK)
    IF (KHECK) 5400, 5400, 5390
5130 CALL CHLT(NVEH,NBLK,KTHLA,KHECK)
    IF (KHECK) 5400, 5400, 5390
5140 CALL CHRT(NVEH,NBLK,KTHLA,KHECK)
    IF (KHECK) 5400, 5400, 5390
5390 CONTINUE
    IF (STPFLG(NVEH).NE.0) GO TO 5395
    NOSTPS(KTHLA) = NOSTPS(KTHLA) + 1
    STPFLG(NVEH) = 1
5395 CONTINUE
    NOWATS(KTHLA) = NOWATS(KTHLA) + 1
5400 CONTINUE
5401 KBLST = FSTBLK(KTHLA) + NOBLKS(KTHLA) - 1
    NVEH = BLKARY(KBLST)
    IF (NVEH.EQ.0) GO TO 5500
    IF (STPFLG(NVEH).NE.999) GO TO 5410
    STPFLG(NVEH) = 0

```



```

      GO TO 5500
5410 KAUX = KBLST -1
      IF (BLKARY(KAUX).NE.0) GO TO 5420
      BLKARY(KAUX) = NVEH
      BLKARY(KBLST) = 0
      STPFLG(NVEH) = 0
      GO TO 5500
5420 CONTINUE
      IF (STPFLG(NVEH).NE.0) GO TO 5425
      NOSTPS(KTHLA) = NOSTPS(KTHLA) + 1
      STPFLG(NVEH) = 1
5425 NOWATS(KTHLA) = NOWATS(KTHLA) + 1
5500 CONTINUE

```

```

C C C C C C C C C C C C C C C C C C C C C C C C C
C      THIS SECTION TERMINATES VECHICLES ON ALL OUTGOING LINKS.
C C C C C C C C C C C C C C C C C C C C C C C C C

```

```

      DO 6100 KTERS=1,NTRMS
      KTHLK = LINTER(KTERS)
      LAN1 = FSTLAN(KTHLK)
      LANL = FSTLAN(KTHLK) + NOLAN(KTHLK) - 1
      DO 6100 KTHLA=LAN1,LANL
      NBLK = FSTBLK(KTHLA)
      NVEH = BLKARY(NBLK)
      IF(NVEH.EQ.0)GO TO 6100
      STPFLG(NVEH)=-999
      BLKARY(NBLK)=0
      NOVEH=NOVEH-1
6100 CONTINUE
      NUMVEH=NOVEST+NOVEIN
      GO TO 2000

```

```

7000 CONTINUE
      NOVEST=NOVEH
      KSTOPS = 0
      KDELAY = 0
      NOVEIN=0
      DO 7200 K=1,KLANES
        KSTOPS = KSTOPS + NOSTPS(K)
        KDELAY = KDELAY + NOWATS(K)
        NOSTPS(K) = 0
        NOWATS(K) = 0
7200 CONTINUE
      STOPS = KSTOPS
      DELAY = FLOAT(KDELAY) * TIMINT
      STOPS=STOPS/FLOAT(NUMVEH)
      DELAY=DELAY/FLOAT(NUMVEH)
      OBJFUN=DELAY+14.5*STOPS
      WRITE(6,141)
141  FORMAT(1H0,12X,3HRUN,5X,6HOBJFUN,7X,5HSTOPS,7X,14HDELAY   NUMVEH)
140  FORMAT(10X,I5,3F12.2,I7)
      WRITE(6,140)KRUNS,OBJFUN,STOPS,DELAY,NUMVEH
      RETURN
      END

```

```

-FOR, IS SNEAK
      SUBROUTINE SNEAK (NXTLNK,NBLK,NVEH,KRN)

```

```

C      THIS SUBROUTINE ALLOWS ONE CAR TO TURN LEFT AS THE
C      LIGHT CHANGES FROM GREEN TO RED. ONLY USED IF VEHICLE

```

```

      INTEGER BLKARY, STPFLG, TRNFLG, FSTLAN, FSTBLK

```

```

COMMON/BLK1/ BLKARY(10000)
COMMON/BLK2/ NTNFLG(2000)
COMMON/BLK3/ STPFLG(2000)
COMMON/BLK4/ TRNFLG(2000)
COMMON/BLK5/ FSTLAN(400)
COMMON/BLK6/ NOLAN(400)
COMMON/BLK7/ PLT(400), PSM(400)
COMMON/BLK8/ FSTBLK(1000)
COMMON/BLK9/ NOBLKS(1000)
RN = RN1(KRN)
IF (RN.LE.PLT(NXTLNK)) GO TO 10
IF (RN.LE.PSM(NXTLNK)) GO TO 20

```

C ALL VEHICLES THAT TAKE THIS PATH WILL TURN RIGHT AT THE NEXT LIGHT

```

NTNFLG(NVEH) = -1
NXTLAN = FSTLAN(NXTLNK)
NXTBK = FSTBLK(NXTLAN) + NOBLKS(NXTLAN) - 1
IF (BLKARY(NXTBK).EQ.0) GO TO 40
GO TO 50
10 CONTINUE

```

C ALL VEHICLES THAT TAKE THIS PATH WILL TURN LEFT AT THE NEXT LIGHT

```

NTNFLG(NVEH) = 1
NXTLAN = FSTLAN(NXTLNK) + NOLAN(NXTLNK) - 1
NXTBK = FSTBLK(NXTLAN) + NOBLKS(NXTLAN) - 1
IF (BLKARY(NXTBK).EQ.0) GO TO 40
GO TO 50
20 CONTINUE

```

C ALL VEHICLES THAT TAKE THIS PATH GO STRAIGHT THROUGH NEXT LIGHT

```

NTNFLG(NVEH) = 0
K1 = NOLAN(NXTLNK)
DO 30 KCHK=1,K1

```

```

      NXTLAN = FSTLAN(NXTLNK) + KCHK - 1
      NXTBK = FSTBLK(NXTLAN) + NOBLKS(NXTLAN) - 1
      IF (BLKARY(NXTBK).EQ.0) GO TO 40
30  CONTINUE
      GO TO 50
40  CONTINUE

```

```

C      VEHICLES THAT ENTER HERE ARE ABLE TO MOVE INTO THE LANE OF THEIR
C      CHOICE ON THE NEXT STREET.

```

```

      BLKARY(NBLK) = 0
      BLKARY(NXTBK) = NVEH
      STPFLG(NVEH) = 999
      TRNFLG(NVEH) = NTNFLG(NVEH)
      NTNFLG(NVEH) = 999
50  CONTINUE
      RETURN
      END

```

```

-FOR, IS CHLT
      SUBROUTINE CHLT (NVEH,NBLK,KTHLAN,KHECK)
      INTEGER BLKARY, STPFLG, FSTBLK
      COMMON/BLK1/ BLKARY(10000)
      COMMON/BLK3/ STPFLG(2000)
      COMMON/BLK8/ FSTBLK(1000)
      NXTLAN = KTHLAN + 1
      NUBKSB = NBLK - FSTBLK(KTHLAN)
      NXTBLK = FSTBLK(NXTLAN) + NUBKSB - 1
      IF (BLKARY(NXTBLK).EQ.0) GO TO 10
      KHECK = 1
      RETURN

```

```

10 CONTINUE
   IF (STPFLG(NVEH)) 20,20,15
15 STPFLG(NVEH) = -1
   KHECK = 1
   RETURN
20 CONTINUE
   BLKARY(NBLK) = 0
   BLKARY(NXTBLK) = NVEH
   STPFLG(NVEH) = 0
   KHECK = -1
   RETURN
   END

```

```

-FOR, IS CHRT
  SUBROUTINE CHRT (NVEH,NBLK,KTHLAN,KHECK)
    INTEGER BLKARY, STPFLG, FSTBLK
    COMMON/BLK1/ BLKARY(10000)
    COMMON/BLK3/ STPFLG(2000)
    COMMON/BLK8/ FSTBLK(1000)
    NXTLAN = KTHLAN - 1
    NUBKSB = NBLK - FSTBLK(KTHLAN)
    NXTBLK = FSTBLK(NXTLAN) + NUBKSB - 1
    IF (BLKARY(NXTBLK).EQ.0) GO TO 10
    KHECK = 1
    RETURN
10 CONTINUE
   IF (STPFLG(NVEH)) 20,20,15
15 STPFLG(NVEH) = -1
   KHECK = 1
   RETURN
20 CONTINUE
   BLKARY(NBLK) = 0

```

```

      BLKARY(NXTBLK) = NVEH
      STPFLG(NVEH) = 0
      KHECK = -1
      RETURN
      END

```

```

-FOR, IS CHKMV
      SUBROUTINE CHKMV (NBLK,NXTLNK,NTRNFG,KHECK,NVEH)

```

```

C      C      C      C      C      C      C      C      C      C      C      C      C      C
C      THIS SUBROUTINE CHECKS TO SEE THAT THE DESIRED STREET      C
C      IS CLEAR, ASSIGN A LANE TO THE ENTERING CAR AND MOVES IT.
C      C      C      C      C      C      C      C      C      C      C      C      C      C

```

```

      INTEGER BLKARY, STPFLG, TRNFLG, FSTLAN, FSTBLK
      COMMON/BLK1/ BLKARY(10000)
      COMMON/BLK2/ NTNFLG(2000)
      COMMON/BLK3/ STPFLG(2000)
      COMMON/BLK4/ TRNFLG(2000)
      COMMON/BLK5/ FSTLAN(400)
      COMMON/BLK6/ NOLAN(400)
      COMMON/BLK8/ FSTBLK(1000)
      COMMON/BLK9/ NOBLKS(1000)
      IF (NTRNFG) 100, 200, 300
100 CONTINUE

```

```

C      THIS IS FOR VEHICLES THAT WILL TURN RIGHT AT THE NEXT INTERSECTION
      NXTLAN = FSTLAN(NXTLNK)

```

```

      NXTBK = FSTBLK(NXTLAN) + NOBLKS(NXTLAN) - 1
      IF (BLKARY(NXTBK) .EQ.0) GO TO 500
      GO TO 400
200  CONTINUE
C
C      THIS IS FOR VEHICLES THAT WILL GO STRAIGHT AT THE NEXT INTERSECTION
C
      K1 = NOLAN(NXTLNK)
      DO 250 KCHK=1,K1
      NXTLAN = FSTLAN(NXTLNK) + KCHK - 1
      NXTBK = FSTBLK(NXTLAN) + NOBLKS(NXTLAN) - 1
      IF (BLKARY(NXTBK) .EQ.0) GO TO 500
250  CONTINUE
      GO TO 400
300  CONTINUE
C
C      THIS IS FOR VEHICLES THAT WILL TURN LEFT AT THE NEXT INTERSECTION
C
      NXTLAN = FSTLAN(NXTLNK) + NOLAN(NXTLNK) - 1
      NXTBK = FSTBLK(NXTLAN) + NOBLKS(NXTLAN) - 1
      IF (BLKARY(NXTBK) .EQ.0) GO TO 500
C
C      DESIRED LANE IS BLOCKED
C
350  STPFLG(NVEH) = -1
400  CONTINUE
      KHECK = 1
      RETURN
500  CONTINUE
C
C      THE VEHICLES ENTERING HERE HAVE FOUND THE DESIRED LANE AND ARE
C      ABLE TO MOVE.
C
      IF (STPFLG(NVEH) .EQ.1) GO TO 350
      BLKARY(NBLK) = 0
      BLKARY(NXTBK) = NVEH

```

```

STPFLG(NVEH) = 999
TRNFLG(NVEH) = NTNFLG(NVEH)
NTNFLG(NVEH) = 999
KHECK = -1
RETURN
END

```

```

-FOR, IS RN1
  FUNCTION RN1(I)
    DIMENSION ARY(200)
    IF (I.NE.1) GO TO 100
    ARY(I)=6734269
    CALL RANDU(ARY,200)
100 RN1 = ARY(I+1)
    IF (I.EQ.199) I=1
    I = I + 1
    RETURN
  END

```

```

-XQT
  13 14. 100
      30.00      0.5      10.0
101 0 20110.753
102 1 20129.412
103-1 202
104 1 20325.641
105 0 20617.544

```


106-1 206
 107 1 205 8.621
 108-1 204
 109 0 20417.241
 110 0 20313.158
 9999
 201 1 101 202 102 204 49 490 510 240
 202 1 201 203 103 205 49 490 510 740
 203 1 202 110 104 206 49 490 510 740
 204 1 109 205 108 201 49 490 510 490
 205 1 204 206 107 202 49 490 510 490
 206 1 105 205 106 203 49 490 510 0
 9999
 101- 201 2 0 0 102 690 202 310 204* 201- 101 2 0
 102- 201 3 0 60 202 710 204 230 101*
 104- 203 3 0 90 110 830 206 80 202*
 105- 206 2 0 120 106 880 205 0 203* 206- 105 2 0
 107- 205 3 0 270 204 430 202 300 206*
 109- 204 2 0 0 201 700 205 300 108* 204- 109 2 0
 110- 203 2 0 150 206 850 202 0 104* 203- 110 2 0
 201- 202 2 430 70 103 930 203 0 205* 202- 201 2 430 190 204 810 101 0 102
 201- 204 3 415 200 205 600 108 200 109*
 202- 103 3 0 *
 202- 203 2 430 000 104 770 110 230 206* 203- 202 2 430 0 205 930 201 70 103
 203- 206 3 415 110 105 580 106 310 205*
 204- 205 2 430 160 202 840 206 0 107* 205- 204 2 430 230 108 770 109 0 201
 204- 108 3 0 *
 205- 202 3 415 220 201 540 103 240 203*
 205- 206 2 430 0 203 480 105 520 106* 206- 205 2 430 000 107 730 204 270 202
 206- 106 3 0 0
 9999
 - FIN

APPENDIX C

The Results of Problems Studied

Problem 1 - Improvement of 6 Variables for AM

Each Row = 6 Offsets

Side Length = 19

$$D_o = \begin{bmatrix} 24 & 74 & 74 & 49 & 49 & 0 \\ 41 & 78 & 78 & 53 & 53 & 4 \\ 28 & 91 & 78 & 53 & 53 & 4 \\ 28 & 78 & 91 & 53 & 53 & 4 \\ 28 & 78 & 78 & 66 & 53 & 4 \\ 28 & 78 & 78 & 53 & 66 & 4 \\ 28 & 78 & 78 & 53 & 53 & 17 \end{bmatrix}$$

Starting Coordinates =

24 74 74 49 49 0

Objective
Function
33.09

Improved Coordinates =

51 60 48 65 63 8

29.93

Number of Simulations Required = 84

Problem 2 - Improvement of 6 Variables for Off-peak

Each Row = 6 Offsets

Side Length = 16

$$D_o = \begin{bmatrix} 24 & 74 & 74 & 51 & 51 & 0 \\ 38 & 77 & 77 & 54 & 54 & 3 \\ 27 & 88 & 77 & 54 & 54 & 3 \\ 27 & 77 & 88 & 54 & 54 & 3 \\ 27 & 77 & 77 & 65 & 54 & 3 \\ 27 & 77 & 77 & 54 & 65 & 3 \\ 27 & 77 & 77 & 54 & 54 & 14 \end{bmatrix}$$

Starting Coordinates =

24 74 74 51 51 0

Objective
Function
37.43

Improved Coordinates =

35 57 73 39 63 9

31.41

Number of Simulations Required = 136

Problem 3 - Improvement of 6 Variables for PM

Each Row = 6 Offsets

Side Length = 11

$$D_o = \begin{bmatrix} 24 & 74 & 74 & 49 & 53 & 0 \\ 34 & 76 & 76 & 51 & 55 & 2 \\ 26 & 84 & 76 & 51 & 55 & 2 \\ 26 & 76 & 84 & 51 & 55 & 2 \\ 26 & 76 & 76 & 59 & 55 & 2 \\ 26 & 76 & 76 & 51 & 63 & 2 \\ 26 & 76 & 76 & 51 & 55 & 10 \end{bmatrix}$$

Starting Coordinates =

24 74 74 49 53 0

Objective
Function
32.46

Improved Coordinates =

35 69 83 55 67 1

27.43

Number of Simulations Required = 43

Problem 4 - Improvement of 7 Variables for AM

Each Row = 6 Offsets, 1 Cycle Time

Side Length = 11

$$D_o = \begin{bmatrix} 24 & 74 & 74 & 49 & 49 & 0 & 49 \\ 34 & 76 & 76 & 51 & 51 & 2 & 51 \\ 26 & 84 & 76 & 51 & 51 & 2 & 51 \\ 26 & 76 & 84 & 51 & 51 & 2 & 51 \\ 26 & 76 & 76 & 59 & 51 & 2 & 51 \\ 26 & 76 & 76 & 51 & 59 & 2 & 51 \\ 26 & 76 & 76 & 51 & 51 & 10 & 51 \\ 26 & 76 & 76 & 51 & 51 & 2 & 59 \end{bmatrix}$$

Starting Coordinates =

24 74 74 49 49 0 49

Objective
Function
33.09

Improved Coordinates =

22 72 69 61 57 4 41

28.79

Number of Simulations Required = 55

Problem 5 - Improvement of 7 Variables for Off-peak

Each Row = 6 Offsets, 1 Cycle Time

Side Length = 11

$$D_o = \begin{bmatrix} 24 & 74 & 74 & 51 & 51 & 0 & 53 \\ 34 & 76 & 76 & 53 & 53 & 2 & 55 \\ 26 & 84 & 76 & 53 & 53 & 2 & 55 \\ 26 & 76 & 84 & 53 & 53 & 2 & 55 \\ 26 & 76 & 76 & 61 & 53 & 2 & 55 \\ 26 & 76 & 76 & 53 & 61 & 2 & 55 \\ 26 & 76 & 76 & 53 & 53 & 10 & 55 \\ 26 & 76 & 76 & 53 & 53 & 2 & 63 \end{bmatrix}$$

Starting Coordinates =

24 74 74 51 51 0 53

Objective
Function
37.43

Improved Coordinates =

29 78 71 68 56 8 38

28.09

Number of Simulations Required = 58

Problem 6 - Improvement of 7 Variables for PM

Each Row = 6 Offsets, 1 Cycle Time

Side Length = 16

$$D_o = \begin{bmatrix} 24 & 74 & 74 & 49 & 53 & 0 & 53 \\ 38 & 77 & 77 & 52 & 56 & 3 & 56 \\ 27 & 88 & 77 & 52 & 56 & 3 & 56 \\ 27 & 77 & 88 & 52 & 56 & 3 & 56 \\ 27 & 77 & 77 & 63 & 56 & 3 & 56 \\ 27 & 77 & 77 & 52 & 67 & 3 & 56 \\ 27 & 77 & 77 & 52 & 56 & 14 & 56 \\ 27 & 77 & 77 & 52 & 56 & 3 & 67 \end{bmatrix}$$

Starting Coordinates =

24 74 74 49 53 0 53

Objective
Function

32.46

Improved Coordinates =

31 66 82 57 54 3 37

25.08

Number of Simulations Required = 40

Problem 7 - Improvement of 12 Variables for AM

Each Row = 6 Green Splits, 6 Offsets

Side Length = 17

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 49 & 49 & 0 \\ 66 & 54 & 54 & 54 & 54 & 54 & 27 & 77 & 77 & 52 & 52 & 3 \\ 54 & 66 & 54 & 54 & 54 & 54 & 27 & 77 & 77 & 52 & 52 & 3 \\ 54 & 54 & 66 & 54 & 54 & 54 & 27 & 77 & 77 & 52 & 52 & 3 \\ 54 & 54 & 54 & 66 & 54 & 54 & 27 & 77 & 77 & 52 & 52 & 3 \\ 54 & 54 & 54 & 54 & 66 & 54 & 27 & 77 & 77 & 52 & 52 & 3 \\ 54 & 54 & 54 & 54 & 54 & 66 & 27 & 77 & 77 & 52 & 52 & 3 \\ 54 & 54 & 54 & 54 & 54 & 54 & 39 & 77 & 77 & 52 & 52 & 3 \\ 54 & 54 & 54 & 54 & 54 & 54 & 27 & 89 & 77 & 52 & 52 & 3 \\ 54 & 54 & 54 & 54 & 54 & 54 & 27 & 77 & 89 & 52 & 52 & 3 \\ 54 & 54 & 54 & 54 & 54 & 54 & 27 & 77 & 77 & 64 & 52 & 3 \\ 54 & 54 & 54 & 54 & 54 & 54 & 27 & 77 & 77 & 52 & 64 & 3 \\ 54 & 54 & 54 & 54 & 54 & 54 & 27 & 77 & 77 & 52 & 52 & 15 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 49 49 0

Objective

Function

33.09

Improved Coordinates =

54 50 53 53 54 54 27 77 76 63 52 3

28.00

Number of Simulations Required = 21

Problem 8 - Improvement of 12 Variables for Off-peak

Each Row = 6 Green Splits, 6 Offsets

Side Length = 14

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 51 & 51 & 0 \\ 63 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 \\ 53 & 63 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 \\ 53 & 53 & 63 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 \\ 53 & 53 & 53 & 63 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 \\ 53 & 53 & 53 & 53 & 63 & 53 & 26 & 76 & 76 & 53 & 53 & 2 \\ 53 & 53 & 53 & 53 & 53 & 63 & 26 & 76 & 76 & 53 & 53 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 36 & 76 & 76 & 53 & 53 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 86 & 76 & 53 & 53 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 86 & 53 & 53 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 63 & 53 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 63 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 12 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 51 51 0

Objective
Function

37.43

Improved Coordinates =

59 52 51 47 63 56 27 74 73 53 63 0

32.17

Number of Simulations Required = 85

Problem 9 - Improvement of 12 Variables for PM

Each Row = 6 Green Splits, 6 Offsets

Side Length = 14

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 49 & 53 & 0 \\ 63 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 \\ 53 & 63 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 \\ 53 & 53 & 63 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 \\ 53 & 53 & 53 & 63 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 \\ 53 & 53 & 53 & 53 & 63 & 53 & 26 & 76 & 76 & 51 & 55 & 2 \\ 53 & 53 & 53 & 53 & 53 & 63 & 26 & 76 & 76 & 51 & 55 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 36 & 76 & 76 & 51 & 55 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 86 & 76 & 51 & 55 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 86 & 51 & 55 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 61 & 55 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 65 & 2 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 12 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 49 53 0

Objective
Function
32.46

Improved Coordinates =

61 49 62 47 63 49 26 78 80 52 52 3

26.80

Number of Simulations Required = 123

Problem 10 - Improvement of 13 Variables for AM

Each Row = 6 Green Splits, 6 Offsets, 1 Cycle Time

Side Length = 10.5

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 49 & 49 & 0 & 49 \\ 60 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 \\ 53 & 60 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 \\ 53 & 53 & 60 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 \\ 53 & 53 & 53 & 60 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 \\ 53 & 53 & 53 & 53 & 60 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 \\ 53 & 53 & 53 & 53 & 53 & 60 & 26 & 76 & 76 & 51 & 51 & 2 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 33 & 76 & 76 & 51 & 51 & 2 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 83 & 76 & 51 & 51 & 2 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 83 & 51 & 51 & 2 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 58 & 51 & 2 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 58 & 2 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 9 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 58 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 49 49 0 49

Objective
Function
33.09

Improved Coordinates =

52 52 47 52 53 49 27 74 72 53 51 3 39

28.26

Number of Simulations Required = 140

Problem 11 - Improvement of 13 Variables for Off-peak

Each Row = 6 Green Splits, 6 Offsets, 1 Cycle Time

Side Length = 10.5

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 51 & 51 & 0 & 53 \\ 60 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 & 55 \\ 53 & 60 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 & 55 \\ 53 & 53 & 60 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 & 55 \\ 53 & 53 & 53 & 60 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 & 55 \\ 53 & 53 & 53 & 53 & 60 & 53 & 26 & 76 & 76 & 53 & 53 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 60 & 26 & 76 & 76 & 53 & 53 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 33 & 76 & 76 & 53 & 53 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 83 & 76 & 53 & 53 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 83 & 53 & 53 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 60 & 53 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 60 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 9 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 53 & 53 & 2 & 62 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 51 51 0 53

Objective
Function
37.43

Improved Coordinates =

48 51 54 53 55 54 30 71 71 57 55 3 48

30.48

Number of Simulations Required = 140

Problem 12 - Improvement of 13 Variables for PM

Each Row = 6 Green Splits, 6 Offsets, 1 Cycle Time

Side Length = 14

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 49 & 53 & 0 & 53 \\ 63 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 \\ 53 & 63 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 \\ 53 & 53 & 63 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 \\ 53 & 53 & 53 & 63 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 \\ 53 & 53 & 53 & 53 & 63 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 63 & 26 & 76 & 76 & 51 & 55 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 36 & 76 & 76 & 51 & 55 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 86 & 76 & 51 & 55 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 86 & 51 & 55 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 61 & 55 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 65 & 2 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 12 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 65 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 49 53 0 53

Objective
Function
32.46

Imprived Coordinates =

59 48 55 54 52 45 28 73 76 55 52 5 44

24.32

Number of Simulations Required = 105

Problem 13 - Improvement of 18 Variables for AM

Each Row = 6 Green Splits, 6 Offsets, 6 Cycle Times

Side Length = 13

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 49 & 49 & 0 & 49 & 49 & 49 & 49 & 49 & 49 \\ 62 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 62 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 62 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 62 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 62 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 62 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 35 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 85 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 85 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 60 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 60 & 2 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 11 & 51 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 60 & 51 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 60 & 51 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 60 & 51 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 60 & 51 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 60 & 51 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 51 & 2 & 51 & 51 & 51 & 51 & 51 & 60 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 49 49 0 49 49 49 49 49 49

Objective Function = 33.09

Improved Coordinates =

57 55 59 57 54 57 19 76 77 53 53 3 50 55 49 48 51 48

Objective Function = 31.22

Number of Simulations Required = 159

Problem 14 - Improvement of 18 Variables for Off-peak

Each Row = 6 Green Splits, 6 Offsets, 6 Cycle Times

Side Length = 12

D ₀ =	51	51	51	51	51	51	24	74	74	51	51	0	53	53	53	53	53
	61	53	53	53	53	53	26	76	76	53	53	2	55	55	55	55	55
	53	61	53	53	53	53	26	76	76	53	53	2	55	55	55	55	55
	53	53	61	53	53	53	26	76	76	53	53	2	55	55	55	55	55
	53	53	53	61	53	53	26	76	76	53	53	2	55	55	55	55	55
	53	53	53	53	61	53	26	76	76	53	53	2	55	55	55	55	55
	53	53	53	53	53	61	26	76	76	53	53	2	55	55	55	55	55
	53	53	53	53	53	53	34	76	76	53	53	2	55	55	55	55	55
	53	53	53	53	53	53	26	84	76	53	53	2	55	55	55	55	55
	53	53	53	53	53	53	26	76	84	53	53	2	55	55	55	55	55
	53	53	53	53	53	53	26	76	76	61	53	2	55	55	55	55	55
	53	53	53	53	53	53	26	76	76	53	61	2	55	55	55	55	55
	53	53	53	53	53	53	26	76	76	53	53	10	55	55	55	55	55
	53	53	53	53	53	53	26	76	76	53	53	2	63	55	55	55	55
	53	53	53	53	53	53	26	76	76	53	53	2	55	63	55	55	55
	53	53	53	53	53	53	26	76	76	53	53	2	55	55	63	55	55
	53	53	53	53	53	53	26	76	76	53	53	2	55	55	55	63	55
	53	53	53	53	53	53	26	76	76	53	53	2	55	55	55	55	63
	53	53	53	53	53	53	26	76	76	53	53	2	55	55	55	55	63

Starting Coordinates =

51 51 51 51 51 51 24 74 74 51 51 0 53 53 53 53 53 53

Objective Function = 37.43

Improved Coordinates =

58 50 56 52 55 47 28 77 72 53 53 7 56 54 54 56 57 50

Objective Function = 31.26

Number of Simulations Required = 101

Problem 15 - Improvement of 18 Variables for PM

Each Row = 6 Green Splits, 6 Offsets, 6 Cycle Times

Side Length = 16

$$D_o = \begin{bmatrix} 51 & 51 & 51 & 51 & 51 & 51 & 24 & 74 & 74 & 49 & 53 & 0 & 53 & 53 & 53 & 53 & 53 \\ 64 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 64 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 64 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 64 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 64 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 64 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 37 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 87 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 87 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 62 & 55 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 66 & 2 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 13 & 55 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 66 & 55 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 66 & 55 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 66 & 55 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 66 & 55 \\ 53 & 53 & 53 & 53 & 53 & 53 & 26 & 76 & 76 & 51 & 55 & 2 & 55 & 55 & 55 & 55 & 66 \end{bmatrix}$$

Starting Coordinates =

51 51 51 51 51 51 24 74 74 49 53 0 53 53 53 53 53

Objective Function = 32.46

Improved Coordinates =

55 55 53 51 55 55 28 78 66 53 57 4 53 52 57 53 57

Objective Function = 27.68

Number of Simulations Required = 131

BIBLIOGRAPHY

Literature Cited

1. John E. Baerwald, ed., Traffic Engineering Handbook, 3d ed., Institute of Traffic Engineers, Washington, D. C., 1965.
2. Stephen B. Miller and John D. C. Little, "Evaluation and Improvement of Traffic Signal Settings by Simulation," Highway Research Board, Record 170, Highway Research Board, National Academy of Sciences, Washington, D. C., 1967.
3. Jesse G. Schwartz, An Urban Street Network Computer Simulation Model, Joint Research Project in Cooperation with the Commonwealth of Massachusetts Department of Public Works, Civil Engineering Systems Laboratory, Massachusetts Institute of Technology, Cambridge, Mass., September 1966.
4. W. Spindley, G. R. Hext and F. R. Himsforth, "Sequential Application of Simplex Designs in Optimization and Evolutionary Operation," Technometrics, Vol. 4, No. 4, November 1962.
5. A. J. Miller, "A Computer Control System for Traffic Networks," Proceedings of the Second International Symposium on the Theory of Road Traffic Flow, 2nd Edition, London 1963, The Organization for Economic Cooperation and Development, Paris, 1965.
6. M. I. Weinberg, H. Goldstein, T. J. McDade and R. H. Wahlen, "Digital-Computer-Controlled Traffic Signal System for a Small City," National Cooperative Highway Research Program Report 29, in conjunction with the Highway Research Board, National Academy of Sciences, Washington, D. C., 1966.
7. Frederick A. Wagner and Daniel L. Gerlough, "Application of Computer Simulation to the Study of Traffic Signal System Operation," Analysis and Control of Traffic Flow Symposium, Detroit, Jan. 9-10, 1968 Proceedings, SAE Paper 680168, 1968.
8. Roy L. Wilshire, Computer Traffic Control, Office of Traffic and Planning, Wichita Falls, Texas, no date.
9. J. N. Bairstow, "What's Wrong with Urban Traffic Control?" Control Engineering, Vol. 16, No. 2, February 1969.

10. Hal G. La Batt, Proposal for Traffic Control System for City of Charleston, S. C., Automatic Signal Division, Laboratory for Electronics, Inc., East Norwalk, Conn., no date.
11. R. Rangarajan and R. M. Oliver, "Allocations of Servicing Periods That Minimize Average Delay for N Time-Shared Traffic Streams," Transportation Science, Vol. 1, No. 2, May 1967.
12. John A. Hillier and Richard Rothery, "The Synchronization of Traffic Signals for Minimum Delay," Transportation Science, Vol. 1, No. 2, May 1967.
13. D. Longley, "A Control Strategy for a Congested Computer-Controlled Traffic Network," Transportation Research, Vol. 2, 1968.
14. John D.C. Little, "The Synchronization of Traffic Signals by Mixed Integer Linear Programming," Operations Research, Vol. 14, No. 4, July-August 1966.
15. D. I. Robertson, Transyt: A Traffic Network Study Tool, Report LR 253, Road Research Laboratory, Crowthorne, England, 1969.
16. M.J.D. Powell, "An Efficient Method for Finding the Minimum of a Function of Several Variables Without Calculatory Deviation," Computer Journal, Vol. 7, 1964.
17. R. Fletcher and M.J.D. Powell, "A Rapidly Convergent Descent Method for Minimization," Computer Journal, Vol. 6, 1963.
18. R. Hooke and T. A. Jeeves, "Direct Search Solution of Numerical and Statistical Problems," Association for Computer Math Journal, Vol. 8, 1961.
19. George E.P. Box and Normal R. Draper, Evolutionary Operation, John Wiley and Sons, Inc., New York, 1969.
20. Bruce R. Glenn, Some Optimization Procedures Used in Response Surface Methodology, Masters Thesis, Virginia Polytechnic Institute, Blacksburg, Va., May 1967.
21. B. H. Carpenter and H. C. Sweeny, "Process Improvement with 'Simplex' Self-Directing Evolutionary Operation," Chemical Engineering, July 5, 1965.
22. Douglas C. Montgomery, Some Techniques for Determining Machine Center Capabilities in a Job Shop, Doctor of Philosophy Thesis, Virginia Polytechnic Institute, Blacksburg, Va. April 1969.
23. J. A. Nelder and R. Mead, "A Simple Method for Function Minimization," Computer Journal, Vol. 7, 1965.

24. George E.P. Box, "Evolutionary Operation: A Method for Increasing Industrial Productivity," Paper presented to the International Conference on Statistical Quality Control, Paris, July 1955, Applied Statistics, VI, 1957.
25. L. T. Fan, C. L. Hwang and F. A. Tillman, "A Sequential Simplex Pattern Search Solution to Production Planning Problems," American Institute of Industrial (AIIE) Engineers Transactions, September 1969.

OTHER REFERENCES

Bone, Alexander J., The Selection of a Cycle Length for Fixed Time Traffic Signals, Systems Laboratory, Department of Civil Engineering, Massachusetts Institute of Technology, Cambridge, Mass., April 1964.

Edie, L. C., "Discussion of Traffic Stream Measurements and Definitions," Proceedings of the Second International Symposium on the Theory of Road Traffic Flow, 2nd Edition, London 1963, The Organization for Economic Cooperation and Development, Paris, 1965.

Gazis, D. C. and Potts, R. B., "The Over-Saturated Intersection," Proceedings of the Second International Symposium on the Theory of Road Traffic Flow, 2nd Edition, London 1963, The Organization for Economic Cooperation and Development, Paris, 1965.

Webster, F. V., "Traffic Signal Settings," Road Research, Technical Paper No. 39, Her Majesty's Stationery Office, London, 1958.